

Vysoká škola báňská - Technická univerzita Ostrava  
Fakulta strojní  
Katedra aplikované mechaniky



Diplomová práce

**Analýza bezpečnostního pásu v leteckých sedadlech**  
*Analysis of Seat Belt of Aircraft Seats*

*Bc. Marek Klemenc*

Vedoucí práce: Ing. Pavel Maršálek

Obor: Aplikovaná mechanika, 3901T003

Ostrava 2016



# Zadání diplomové práce

Student:

**Bc. Marek Klemenc**

Studijní program:

N2301 Strojní inženýrství

Studijní obor:

3901T003 Aplikovaná mechanika

Téma:

**Analýza bezpečnostního pásu v leteckých sedadlech**  
**Analysis of Seat Belt of Aircraft Seats**

Jazyk vypracování:

čeština

Zásady pro vypracování:

1. Rešerše problematiky testování
2. Analýza bezpečnostních pásů
3. Studium explicitní metody řešení diferenciálních rovnic
4. Simulace dynamického chování bezpečnostních pásů během testování
5. Zhodnocení a závěr

Seznam doporučené odborné literatury:

- [1] SAE AS 8049: Performance Standard for Seats in Civil Rotorcraft, Transport Aircraft, and General Aviation Aircraft.  
[2] Wriggers, P.: Nonlinear Finite Element Methods; Prentice Hall; Watertown, MA, USA, 2008; ISBN: 978-3-540-71001.  
[3] LSTC: LS-DYNA Theory Manual; <http://www.lstc.com/>.  
[4] LSTC: LS-DYNA Keyword User's Manual Vol. I; <http://www.lstc.com/>.  
[5] LSTC: LS-DYNA Keyword User's Manual Vol. II; <http://www.lstc.com/>.  
[6] LSTC: LS-DYNA Keyword User's Manual Vol. III; <http://www.lstc.com/>.

Formální náležitosti a rozsah diplomové práce stanoví pokyny pro vypracování zveřejněné na webových stránkách fakulty.

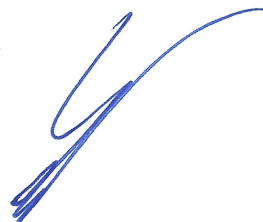
Vedoucí diplomové práce: **Ing. Pavel Maršálek**

Datum zadání: 11.12.2015

Datum odevzdání: 16.05.2016



doc. Ing. Radim Halama, Ph.D.  
vedoucí katedry



doc. Ing. Ivo Hlavatý, Ph.D.  
děkan fakulty





## Prohlášení

Prohlašuji, že jsem celou diplomovou práci včetně příloh vypracoval samostatně pod vedením vedoucího diplomové práce a použil jsem pouze podklady uvedené v příloženém seznamu. Nemám závažný důvod proti užití tohoto školního díla ve smyslu §60 Zákona č. 121/2000 Sb., o právu autorském, o právech souvisejících s právem autorským a o změně některých zákonů (autorský zákon).

V Hati dne 16. května 2016

*marc Klemens*



Prohlašuji, že

- jsem byl seznámen s tím, že na moji diplomovou práci se plně vztahuje zákon č. 121/2000 Sb., autorský zákon, zejména § 35 – užití díla v rámci občanských a náboženských obřadů, v rámci školních představení a užití díla školního a § 60 – školní dílo.
- беру на ве́домі́, že Vysoká škola báňská – Technická univerzita Ostrava (dále jen „VŠB-TUO“) má právo nevýdělečně ke své vnitřní potřebě diplomovou práci užít (§ 35 odst. 3).
- souhlasím s tím, že diplomová práce bude v elektronické podobě uložena v Ústřední knihovně VŠB-TUO k nahlédnutí a jeden výtisk bude uložen u vedoucího diplomové práce. Souhlasím s tím, že údaje o kvalifikační práci budou zveřejněny v informačním systému VŠB-TUO.
- bylo sjednáno, že s VŠB-TUO, v případě zájmu z její strany, uzavřu licenční smlouvu s oprávněním užít dílo v rozsahu § 12 odst. 4 autorského zákona.
- bylo sjednáno, že užít své dílo – diplomovou práci nebo poskytnout licenci k jejímu využití mohu jen se souhlasem VŠB-TUO, která je oprávněna v takovém případě ode mne požadovat přiměřený příspěvek na úhradu nákladů, které byly VŠB-TUO na vytvoření díla vynaloženy (až do jejich skutečné výše).
- беру на ве́домі́, že odevzdáním své práce souhlasím se zveřejněním své práce podle zákona č. 111/1998 Sb., o vysokých školách a o změně a doplnění dalších zákonů (zákon o vysokých školách), ve znění pozdějších předpisů, bez ohledu na výsledek její obhajoby

V Hati dne 16. května 2016



.....  
podpis

Jméno a příjmení autora práce:

Marek Klemenc

Adresa trvalého pobytu autora práce:

Lipová 187, Hať



## Anotace diplomové práce

KLEMENC, M. *Analýza bezpečnostního pásu v leteckých sedadlech: diplomová práce*. Ostrava: VŠB - Technická univerzita Ostrava, Fakulta strojní, Katedra aplikované mechaniky, 2016, 135 s. Vedoucí práce: Maršálek, P.

Diplomová práce se zabývá matematickým modelováním nárazové zkoušky leteckého sedadla pro stanovení časového průběhu reakcí v kotevních bodech pánevního bezpečnostního pásu a jejich maximálních hodnot. Práce byla zadána na základě požadavků praxe společnosti IDEA AIR s.r.o. a její výsledky budou zohledněny při vývoji nového typu leteckých sedadel. V první části práce je provedena rešerše problematiky testování leteckých sedadel a rešerše používaných typů bezpečnostních pásů a jejich materiálů. Následující část se zabývá matematickým pozadím modelování dynamických úloh pomocí metody konečných prvků (MKP). Odvozené postupy jsou dále verifikovány na několika příkladech a implementovány v programovacím jazyce Python. Vlastní výpočetní algoritmus je v závěrečné části práce aplikován na analýzu bezpečnostního pásu a výsledky jsou srovnány se simulací úlohy v komerčním softwaru LS-DYNA.

## Annotation of master thesis

KLEMENC, M. *Analysis of Seat Belt of Aircraft Seats: master thesis*. Ostrava: VŠB - Technical University of Ostrava, Faculty of Mechanical Engineering, Department of Applied Mechanics, 2016, 135 p. Head of thesis: Maršálek, P.

This thesis deals with mathematical modeling of crash test aircraft seats. The main goal is to determine time course of the reactions in anchoring points of lap belt and their maximum values. The thesis was awarded on the basis of practical requirements of company IDEA AIR s.r.o. Results will be reflected in development of a new type aircraft seats. The first part deals with the testing of aircraft seats, types of seat belts and their materials. The next part deals with the mathematical modeling of dynamic problems using the finite element method (FEM). Derived procedures are further verified in several examples and implemented in the Python programming language. Own calculation algorithm is applied to the analysis of safety belt in the final part. Results are compared with the simulation performed in commercial software LS-DYNA.



# Obsah

<b>Seznam použitých značek a symbolů</b>	<b>13</b>
<b>1 Rešerše problematiky testování</b>	<b>19</b>
1.1 Testování sedadel . . . . .	20
1.1.1 Konfigurace dynamických testů . . . . .	20
1.1.2 Test #1 - Vyhodnocení strukturální integrity sedadla . . . . .	21
1.1.3 Test #2 - Vyhodnocení integrity struktury sedadla a chování bezpečnostního pásu . . . . .	21
1.2 Testovací figurína . . . . .	22
<b>2 Analýza bezpečnostních pásů</b>	<b>25</b>
2.1 Typy pásů . . . . .	25
2.2 Materiály popruhové tkaniny pásů . . . . .	26
2.3 Geometrie bezpečnostního pásu . . . . .	27
2.4 Bezpečnostní spony . . . . .	29
<b>3 Explicitní metoda řešení pohybových rovnic</b>	<b>31</b>
3.1 Odvození explicitní metody . . . . .	32
3.1.1 Startovací procedura . . . . .	33
3.1.2 Časový krok . . . . .	34
3.2 Aplikace explicitní metody . . . . .	35
3.2.1 Diskrétní systém . . . . .	37
3.2.1a Příklad volného netlumeného kmitání s 1 stupněm volnosti . . . . .	37
3.2.1b Příklad volného tlumeného kmitání s 1 stupněm volnosti . . . . .	39
3.2.2 Kontinuum - maticová formulace . . . . .	42
3.2.2a Příklad kmitání struny . . . . .	49
3.2.2b Příklad kmitání systému prut-pružina . . . . .	53
3.2.2c Kmitání dvojitého kyvadla . . . . .	56
<b>4 Simulace dynamického chování bezpečnostních pásů</b>	<b>67</b>
4.1 Analýza bezpečnostního pásu pomocí vlastního algoritmu . . . . .	69
4.1.1 Model pásu . . . . .	69
4.1.2 Rozložení hmotnosti figuríny do uzlů pásu . . . . .	71
4.1.3 Okrajové a počáteční podmínky . . . . .	72
4.1.4 Výsledky analýzy . . . . .	73
4.2 Analýza bezpečnostního pásu pomocí LS-DYNA . . . . .	81

4.2.1	Popis modelu . . . . .	81
4.2.2	Model sedadla a podlahy . . . . .	82
4.2.3	Model pásu . . . . .	83
4.2.4	Model figuríny . . . . .	84
4.2.5	Definice kontaktů . . . . .	85
4.2.6	Výsledky analýzy . . . . .	86
<b>5</b>	<b>Závěr a zhodnocení</b>	<b>89</b>
5.1	Zjednodušený 2D model . . . . .	90
5.2	3D model . . . . .	91
5.3	Zhodnocení a doporučení . . . . .	92
5.4	Využití výsledků . . . . .	93
5.5	Pokračování práce . . . . .	94
<b>A</b>	<b>Diskrétní systém</b>	<b>97</b>
<b>B</b>	<b>Struna</b>	<b>99</b>
<b>C</b>	<b>Prut-pružina</b>	<b>103</b>
<b>D</b>	<b>Kyvadlo</b>	<b>109</b>
D.1	Numerické řešení analytických diferenciálních rovnic . . . . .	109
D.2	Numerické řešení explicitní metodou pomocí MKP . . . . .	110
<b>E</b>	<b>Bezpečnostní pás 2D</b>	<b>117</b>
<b>F</b>	<b>CD nosič</b>	<b>123</b>
<b>G</b>	<b>Seznam použitých zkratk</b>	<b>131</b>
	<b>Literatura</b>	<b>133</b>



# Seznam použitých značek a symbolů

Značka	Jednotka	Význam
<b>A</b>	-	matice koeficientů
$A$	$\text{mm}^2$	plocha průřezu
$A_p$	$\text{mm}^2$	plocha průřezu pásu
$B_1, B_2$	mm	rozměr sedadla
<b>C</b>	-	matice tlumení
<b>D</b>	-	matice materiálových konstant
$E$	MPa	modul pružnosti v tahu
<b>F</b>	-	vektor vnějších sil
$F$	N, lb	síla
<b>F<sub>0</sub></b>	-	vektor počátečního vnějšího zatížení
$F_0$	N	počáteční zatížení
$F_D$	N	dovolené zatížení pásu
<b>F<sub>eff</sub></b>	-	vektor neznámých
$F_S$	N	dovolené zatížení spony
$G$	MPa	modul pružnosti ve smyku
$H$	mm	výška pásu
$H_1, H_2$	mm	rozměr sedadla
<b>K</b>	-	matice tuhosti
<b>K<sub>G</sub></b>	-	matice geometrické tuhosti
<b>K<sub>L</sub></b>	-	matice tuhosti elementu
<b>K<sub>T</sub></b>	-	elastická matice tuhosti
$L$	m, in	délka

Značka	Jednotka	Význam
$L_e, L'_e$	m	délka elementu
$L_h$	mm	délka hrany elementu
$L_p$	mm	délka pásu
$L_n$	mm	délka nástavce pásu
<b>M</b>	-	matice hmotnosti
<b>M<sub>eff</sub></b>	-	matice koeficientů
<b>N</b>	-	matice tvarových funkcí
$N$	N	normálová síla
<b>P</b>	-	vektor vnitřních uzlových reakcí
<b>R</b>	-	vektor výsledných reakcí
$R, R_L, R_P$	N	složka vektoru výsledných reakcí
$R_{L\max}, R_{P\max}$	N	maximální výsledná reakce
$R_{Lx}, R_{Ly}, R_{Lz}$	N	složka výsledné reakce
$R_{Px}, R_{Py}, R_{Pz}$	N	složka výsledné reakce
$R'_{Px}, R'_{Py}, R'_{Pz}$	N	složka výsledné reakce
$R_x, R_y, R_{xy}, R_{yz}, R_{xz}$	N	složka výsledné reakce
<b>T</b>	-	transformační matice
$T_{\min}$	s	minimální perioda
$V$	m <sup>3</sup>	objem
$a, a_1, a_2$	m s <sup>-2</sup>	zrychlení
$a_x, a_y$	m s <sup>-2</sup>	zrychlení v ose x a y
<b>b</b>	-	vektor pravé strany
$b$	N m <sup>-1</sup> s	koeficient tlumení
$b_p$	mm	šířka pásu
$d$	mm	průměr
$g$	m s <sup>-2</sup>	tíhové zrychlení
$h$	in	výška
$k$	kg	hmotnostní parametr
$k$	N m <sup>-1</sup>	tuhost

Značka	Jednotka	Význam
$k_p$	$\text{lb in}^{-1}$	tuhost pružiny
$\mathbf{l}, \mathbf{l}'$	-	vektor elementu
$l_1, l_2$	m	délka
$m, m_{z1}, m_{z2}$	kg	hmotnost
$m_e$	kg	hmotnost elementu
$m_f$	kg	hmotnost figuríny
$m_i, \tilde{m}, \bar{m}$	kg	hmotnost v uzlu
$q$	kg	hmotnostní parametr
$q$	$\text{N m}^{-1}$	spojité zatížení
$s_p$	mm	tloušťka pásu
$s_s$	mm	tloušťka elementu
$t, t_{r1}, t_{r2}, t_b, t_c$	s	čas
$\mathbf{u}$	-	vektor posuvů
$\dot{\mathbf{u}}$	-	vektor rychlostí
$\ddot{\mathbf{u}}$	-	vektor zrychlení
$\tilde{\mathbf{u}}$	-	vektor zobecněných uzlových posunutí
$\dot{\tilde{\mathbf{u}}}$	-	vektor zobecněných uzlových rychlostí
$\mathbf{u}_0$	-	vektor počátečních posuvů
$\dot{\mathbf{u}}_0$	-	vektor počátečních rychlostí
$\ddot{\mathbf{u}}_0$	-	vektor počátečních zrychlení
$u_0$	m	počáteční posuv
$\dot{u}_0$	$\text{m s}^{-1}$	počáteční rychlost
$u_1, u_2$	m	složka vektoru zobecněných uzlových posunutí
$u_a$	m	amplituda kmitání
$u_i$	m	složka vektoru posuvů
$u_x, u_y$	m	výchylka v ose x a y
$v_0$	$\text{m s}^{-1}$	počáteční rychlost
$v_1, v_2$	m	složka vektoru zobecněných uzlových posunutí
$v_x, v_y$	$\text{m s}^{-1}$	rychlost v ose x a y

Značka	Jednotka	Význam
$\mathbf{x}$	-	vektor neznámých
$x, y, z$	m	souřadnice
$x_1, x_2, y_1, y_2$	m	souřadnice uzlu
$x'_1, x'_2, y'_1, y'_2$	m	souřadnice uzlu
$\Delta t$	s	časový krok
$\Delta t_{\text{crit}}$	s	kritický časový krok
$\epsilon$	-	vektor poměrných deformací
$\Pi_d$	J	deformační energie
$\Pi_k$	J	kinetická energie
$\Omega$	$\text{s}^{-1}$	vlastní kruhová frekvence tlumeného kmitání
$\Omega_0$	$\text{s}^{-1}$	vlastní kruhová frekvence netlumeného kmitání
$\Omega_{\text{max}}$	$\text{s}^{-1}$	maximální vlastní kruhová frekvence
$\alpha$	-	koefficient vnějšího tlumení
$\alpha_1, \alpha_2$	°	směr zrychlení
$\alpha_p$	°	sklon pásu
$\alpha_R, \alpha_{RL}, \alpha_{RP}, \alpha_{yz}$	°	směr působení reakce
$\alpha_{Lxy}, \alpha_{Pxy}, \alpha_{Lxz}, \alpha_{Pxz}$	°	směr působení reakce
$\beta$	-	koefficient vnitřního tlumení
$\beta_1, \beta_2$	°	úhel sedadla
$\gamma$	-	zkos
$\delta$	$\text{s}^{-1}$	konstanta doznívání
$\epsilon, \epsilon_x$	-	poměrná deformace
$\eta$	-	faktor časového kroku
$\theta, \theta_0, \theta_1, \theta_2$	°	úhel natočení
$\dot{\theta}_1, \dot{\theta}_2$	$\text{s}^{-1}$	úhlová rychlost
$\ddot{\theta}_1, \ddot{\theta}_2$	$\text{s}^{-2}$	úhlové zrychlení
$\mu$	-	Poissonovo číslo
$\nu$	kg	hmotnostní parametr
$\pi$	-	Ludolfovo číslo

Značka	Jednotka	Význam
$\rho$	$\text{kg m}^{-3}$	hustota
$\rho_m$	$\text{kg m}^{-1}$	měrná hmotnost pásu
$\boldsymbol{\sigma}$	-	vektor napětí
$\sigma$	MPa	normálové napětí
$\tau$	MPa	smykové napětí
$\varphi$	$^\circ$	natočení elementu
$\varphi_0$	rad	fázový posuv



# Kapitola 1

## Rešerše problematiky testování

Lidstvo se od samého počátku existence snaží zdokonalovat ve všech oblastech lidského poznání. Výsledky tohoto snažení jsou dobře vidět především v technických oblastech. První nástroje k usnadnění práce používali naši předkové už před několika miliony let. Primitivní nástroje se postupem času zdokonalovaly a už staří Egypťané byli schopni několik tisíc let př.n.l. stavět obrovské pyramidy za pomoci jednoduchých, ale důmyslných mechanismů. Vývoj šel stále rychleji vpřed a na přelomu 18. a 19. století dochází ke skutečné průmyslové revoluci, jejímž symbolem je parní stroj. O sto let později již sjíždí z montážní linky v Detroitu první sériově vyráběný vůz Ford model T. Počátkem 20. století, v roce 1903, se díky bratrům Wrightovým rovněž začíná psát historie motorového létání. Technický pokrok jde stále rychleji kupředu a na dnešní konstruktéry a vývojáře jsou kladeny stále větší nároky. Nově vyvíjené a zdokonalované materiály a výrobní technologie umožňují návrh a výrobu stále lehčích, pevnějších a bezpečnějších konstrukcí a strojních součástí. Současným trendem je rovněž snižování zatížení životního prostředí a výrobních nákladů na minimum.

K vývoji dochází i v oblasti leteckých sedadel. Jednou z firem, která se touto oblastí zabývá, je společnost IDEA AIR s.r.o., která je zadavatelem této diplomové práce.

Ve všech odvětvích průmyslu je potřeba už při návrhu dílů dodržovat určité zásady s ohledem na bezpečnost budoucího provozu. V dopravním průmyslu, obzvláště v leteckém, platí toto dvojnásobně. Za dlouhá léta vývoje byla zavedena řada závazných předpisů a norem, často s celosvětovým dosahem. Pro certifikaci některých výrobků, ať už ke komerčnímu, nebo soukromému využití, je při vývoji nutné tyto nařízení dodržet. V leteckém průmyslu existuje řada organizací, které dohlíží na dodržování norem, provádějí potřebné certifikace, či dané normy vydávají. Mezinárodními standardy v oblasti leteckých sedadel se zabývá norma SAE AS8049C [19], která definuje bezpečnostní a výkonnostní parametry sedadel vrtulníků, dopravních letadel i všech ostatních leteckých dopravních prostředků, které zahrnují např. akrobatická letadla, soukromé letouny apod. Tato práce se zaměřuje na použití bezpečnostního pásu a jeho uložení u sedadla v dopravním letadle pro civilní sféru. Uvedená norma vymezuje důležité pojmy a definuje minimální výkonnostní parametry, kvalifikační požadavky a minimální dokumentaci potřebnou pro sedadla cestujících a sedadla posádky. Předpisy se týkají konstrukce sedadel a jejich příslušenství, včetně bezpečnostních pásů, požadavků na tuhost a pevnost konstrukce nejen statickou, ale i dynamickou. Rovněž předepisuje požadované zátěžové zkoušky spolu s kritérii pro jejich vyhodnocení. Cílem je zajištění odolnosti sedadla, pohodlí pasažéra a především jeho bezpečnosti za normálních provozních podmínek, i při

kritických zátěžných stavech. Norma popisuje požadavky na sedadla orientovaná ve směru a proti směru letu. Pro boční sedadla je nutné zahrnout dodatečné požadavky, jenž předepisuje norma SAE AS8049/1.

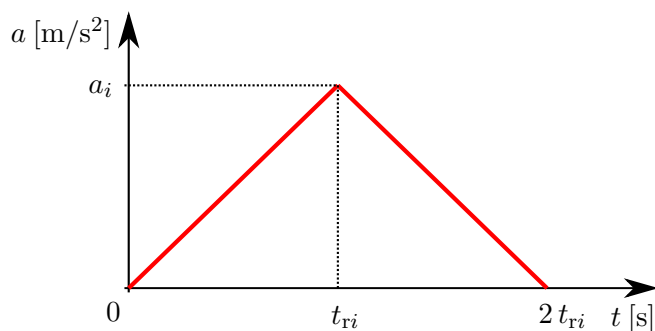
## 1.1 Testování sedadel

Pro posouzení chování sedadel je zapotřebí provést řadu testů. Ty se dělí na statické a dynamické. Statické zátěžové testy mají prokázat schopnost konstrukce sedadla, včetně bezpečnostních pásů i konstrukčních prvků, přenést předepsané limitní zatížení bez porušení struktury po dobu minimálně 3 sekund. Dojde-li k takovému porušení, které by mohlo vést ke snížení bezpečnosti cestujících, nebo by zasahovalo do bezpečného provozu letadla, je zkouška vyhodnocena jako neúspěšná. Trvalé deformace po testu musí rovněž vyhovovat dovoleným hodnotám. Statické testy se provádějí pro polohu sedadla, která odpovídá poloze při rolování letadla po dráze, při vzletu a při přistání.

Druhou skupinou jsou zkoušky dynamické, tzv. crash testy. Ty jsou oproti statickým náročnější z hlediska provedení a vyhodnocení testu a splnění všech požadovaných kritérií. Sedadlo, bezpečnostní prvky, resp. bezpečnostní pás, a další související zařízení musí spolehlivě fungovat jako jeden celek za všech okolností a poskytnout ochranu sedící osobě např. během nárazu při nouzovém přistání letadla, při velkých turbulencích a podobných krizových situacích. Dle mezinárodní normy jsou požadovány minimálně dva dynamické testy pro vyhodnocení chování celku sedadla při nárazu. Praktické provedení testů spočívá v upevnění konstrukce sedadla spolu s testovací figurínou na testovací zařízení, kterým je polohovatelná plošina pohybující se po kolejnicích, uvedení plošiny se sedadlem do pohybu na požadovanou rychlost, či předepsaným zrychlením, a následuje náraz plošiny do bariéry, či zabrždění plošiny tak, aby bylo dosaženo předepsaného zpomalení. V průběhu celého testu probíhá měření potřebných veličin a celý test je zaznamenáván vysokorychlostními kamerami. Požadované nastavení testů je popsáno v následujícím textu.

### 1.1.1 Konfigurace dynamických testů

Norma předepisuje požadovaný charakter zatížení během testů (viz obr. 1.1). Profil testovacího impulsu je stejný pro obě varianty. Konkrétní hodnoty jsou však odlišné a jsou uvedeny u jednotlivých testů.



Obrázek 1.1: Předepsaný průběh zatížení

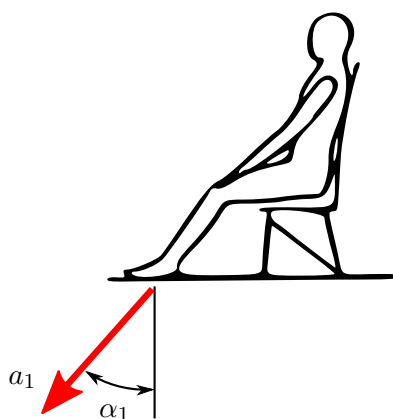


Při reálném testování se zpomalení měří akcelerometry a výsledný průběh se vyhodnocuje dle [19]. Jestliže průběh zrychlení nemá požadovaný tvar (nebo není v toleranci), je test považován za neúspěšný.

### 1.1.2 Test #1 - Vyhodnocení strukturální integrity sedadla

První test slouží k posouzení chování celku sedadla za testovacích podmínek, kdy převládající nárazová síla působí ve směru páteře sedícího pasažéra v kombinaci s čelní nárazovou silou. Tímto testem se posuzuje odolnost struktury sedadla, jeho tlumící vlastnosti, vyhodnocují se průběhy sil působících na bederní páteř cestujícího a zkoumají se trvalé deformace konstrukce. Rovněž mohou být vyhodnoceny průběhy posuvu, rychlosti a zrychlení hlavy testovací figuríny. Konfigurace testu je znázorněna na obr. 1.2. Šipkou je znázorněn požadovaný směr působení setrvačné síly při testu pod úhlem  $\alpha_1 = 30^\circ$ . Naměřené zpomalení dosažené při této zkoušce musí být minimálně  $a_1 = 14g$ , což odpovídá hodnotě  $a_1 \approx 140 \text{ m/s}^2$ . Doba lineárního náběhu, při kterém je dosaženo tohoto zpomalení, nesmí překročit maximální hodnotu  $t_{r1} = 80 \text{ ms}$ .

Jak již bylo zmíněno, tento test slouží především k posouzení pevnosti konstrukce sedadla. **Pro vyhodnocení chování bezpečnostního pásu a reakcí v uchycení pásu nemá prakticky žádný význam a tato práce se jím dále nezabývá.**

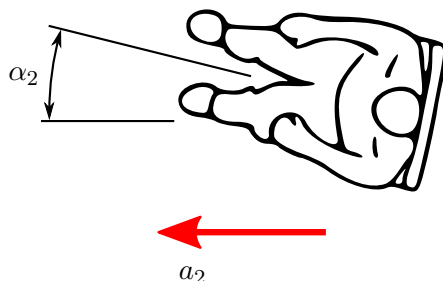


Obrázek 1.2: Konfigurace testu #1

### 1.1.3 Test #2 - Vyhodnocení integrity struktury sedadla a chování bezpečnostního pásu

Druhý test posuzuje chování systému za podmínek, kdy převládající nárazová síla působí ve směru podélné osy letadla v kombinaci s boční nárazovou silou. Tento test rovněž posuzuje odolnost struktury sedadla, její trvalé deformace a dále chování a zatížení pánevního bezpečnostního pásu, případně i hrudního, pokud je použit. Dále je možné vyhodnotit časový průběh trajektorie hlavy testovací figuríny, průběhy kinematických veličin pohybu hlavy (posuv, rychlost, zrychlení) a zatížení konzoly sedadla, která je uložena v kolejnicích nebo je jinak připevněna k podlaze, resp. konstrukci letadla. Konfigurace testu #2 je znázorněna na obr. 1.3. Šipkou je znázorněn požadovaný směr působení setrvačné síly při testu. Naměřené

zpomalení dosažené při této zkoušce musí být minimálně  $a_2 = 16 \text{ g}$ , což odpovídá hodnotě  $a_2 \approx 160 \text{ m/s}^2$ . Doba lineárního náběhu, při kterém je dosaženo tohoto zpomalení nesmí překročit hodnotu  $t_{r2} = 90 \text{ ms}$ .



Obrázek 1.3: Konfigurace testu #2

**Tento test je klíčový pro vyhodnocení průběhu zatížení v místech uložení bezpečnostních pásů.** Navíc je zde před provedením testu požadováno natočení sedadla o minimální úhel  $\alpha_2 = 10^\circ$  vzhledem k podélné ose letadla. Mimo to je před testem předepsána požadovaná deformace podlahy, resp. konstrukce, ve které je sedadlo při zkoušce uloženo. Tato deformace má prokázat, že celý systém sedadla zůstane během nárazu i po něm plně funkční a nedojde k porušení upevnění k podlaze, ani v případě silně zdeformované struktury letadla (nebo konstrukce sedadla). Při fyzickém testování je sedadlo zpravidla uloženo na tuhých stavitelných nosnících a požadovaná deformace podlahy se realizuje natočením těchto nosníků. Pro vyhodnocení průběhu reakcí v kotevních bodech pásu však tato podmínka nemá význam a nebude proto v pozdějších simulacích uvažována. Samotná podlaha nemusí být při testu #2 zahrnuta. V případě umístění sedadel ve více řadách je třeba testovat alespoň dvě řady sedadel za sebou v odpovídající vzdálenosti. Při testu se pak navíc vyhodnocují kritéria poranění nohou a hlavy sedící osoby. Pro prvotní návrh, kdy není známo rozmístění sedadel ani přesná konstrukce sedadla, je však tato podmínka nevýznamná.

## 1.2 Testovací figurína

Pro oba testy se používá standardizovaná, kalibrovaná testovací figurína (Dummy, ATD) s modelovým označením Hybrid III [20]. Tyto figuríny simulují lidskou odezvu na vnější fyzikální podněty a slouží k testování možnosti vzniku poranění osob a chování lidského těla např. při nárazových zkouškách v automobilovém, vlakovém či leteckém průmyslu nebo pro potřeby armády. Figuríny jsou osazeny snímači kinematických a silových veličin a v průběhu testu zaznamenávají údaje, pomocí nichž se následně vyhodnocují kritéria poranění a další požadované informace. Parametry figuríny (hmotnost, rozměry) se řídí příslušnými normami, viz [19]. Označení Hybrid III zahrnuje několik velikostí testovacích figurín. Jedná se o 50 a 95 percentilovou mužskou figurínu, 5 percentilovou ženskou figurínu a figuríny deseti, šesti a tříletého dítěte. Pro výše uvedené testy, stejně jako pro převážnou část všech prováděných crash testů v různých odvětvích, se používá 50 percentilová mužská figurína o hmotnosti  $m_f = 77 \text{ kg}$  (obr. 1.4).



Obrázek 1.4: 50 percentilová figurína Hybrid III [21]

Kromě zmíněných typů existují i další speciální figuríny, určené např. pro testování bočních nárazů, testování bezpečnostních pásů bez možnosti vyhodnocení kritérií poranění, apod. Většinu testovacích figurín zpřístupňuje firma LSTC na základě platné licence ve formě odladěného a kalibrovaného konečnoprvkového modelu.



## Kapitola 2

# Analýza bezpečnostních pásů

V předchozí kapitole byly stanoveny parametry testování sedadel, které jsou výchozí rovněž pro numerické simulace. Dále je potřeba stanovit konkrétní typ bezpečnostního pásu, použitý materiál a geometrii, resp. rozměry bezpečnostního pásu.

### 2.1 Typy pásů

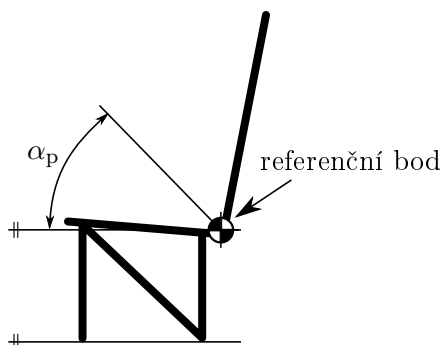
V letectví se používá několik typů bezpečnostní pásů. Liší se především podle oblastí jejich použití. Pro posádku dopravních letadel jsou určeny pásy vícebodové, tj. tříbodové, čtyřbodové nebo pětibodové, pro pasažéry ekonomické třídy pak nejčastěji pásy dvoubodové. V soukromých a sportovních letadlech, vrtulnících či vojenských nebo jiných speciálních letounech je většinou vyžadováno použití vícebodových pásů. Některé vícebodové pásy mohou být vybaveny samonavíjecím zařízením, podobně jako je tomu u automobilového pásu. Schématické znázornění nejběžnějších typů je na obr. 2.1. Tato práce je zaměřena na analýzu **dvoubodového bezpečnostního pásu**, který obepíná pouze pánev sedícího pasažéra.



Obrázek 2.1: Schéma bezpečnostních pásů; zleva: 2 bodový; Y-pás; 3 bodový; 4 bodový; 5 bodový

Z konstrukčního hlediska má být dle normy [19] pánevní dvoubodový pás navržen tak, aby vertikální úhel mezi střednicí pásu a vodorovnou čarou procházející referenčním bodem sedadla byl v rozmezí  $\alpha_p = 30^\circ - 55^\circ$  (viz obr. 2.2). Kotevní body pánevního pásu navíc nesmí být více než 51 mm před referenčním bodem sedadla. Účelem správné polohy uložení pásů je zajistit, aby v průběhu statických ani dynamických zkoušek nedošlo ke sklouznutí

pásu z pánve pasažéra. Při návrhu je potřeba rovněž vyloučit možnost rozřezání či jiného porušení pásu o jinou součást sedadla, což by mohlo vést k fatálnímu snížení bezpečnosti.



Obrázek 2.2: Referenční bod sedadla

## 2.2 Materiály popruhové tkaniny pásů

Nejčastěji používané materiály popruhové tkaniny, ze které je pás vyroben, jsou nylon nebo polyester. V [4] je provedena parametrická studie a srovnání výsledků pro oba materiály. Z článku vyplývá, že při simulacích se maximální hodnoty sledovaných veličin, tj. reakcí v kotevních bodech pásu, pro oba materiály příliš neliší. Dle literatury se jeví jako bezpečnější nylonový pás a bude v této práci uvažován. Pro stanovení materiálového modelu, tj. závislosti zátěžné síly na poměrném prodloužení pásu, je zapotřebí provést tahovou zkoušku, jak popisuje např. [5]. Tato materiálová charakteristika byla odečtena z [4]. Původní materiálová odezva byla linearizována a výsledná závislost poměrného prodloužení pásu na zatěžující síle je patrná z obr. 2.3. V tab. 2.1 jsou uvedeny souřadnice bodů stanovené materiálové charakteristiky. Pro srovnání je vykreslena rovněž linearizovaná materiálová křivka polyesterového pásu a její hodnoty jsou uvedeny v tab. 2.2.

U nylonových pásů dosahuje dovolené zatížení v tahu hodnoty  $F_D = 13,3 \text{ kN}$ . To je dáno především pevností spony pásu. Samotná nylonová popruhová tkanina dosahuje únosnosti až 24 kN, polyesterová tkanina až 35 kN.

Všechny použité materiály musí dále splňovat požadavky na **nehořlavost**.

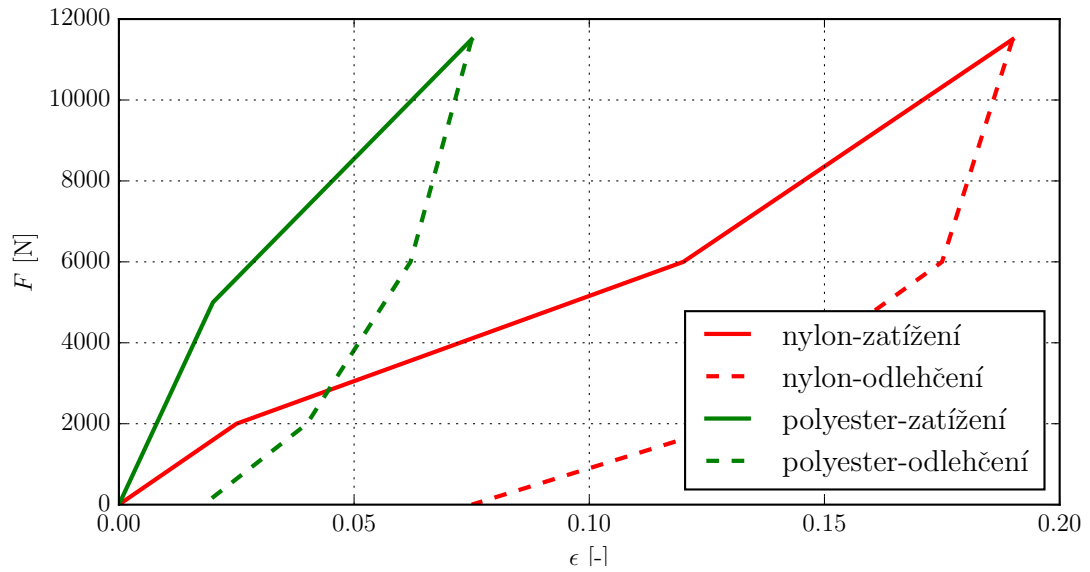
F [N]	2000	6000	11 500	6000	1800	0
$\varepsilon$ [-]	0,025	0,12	0,19	0,175	0,125	0,075

Tabulka 2.1: Souřadnice materiálové charakteristiky nylonového pásu

Pro pozdější analýzy je potřeba určit modul pružnosti nylonového pásu, který je určen z linearizovaného průběhu zátěžné větve materiálové křivky nylonového pásu, viz obr. 2.4.

Modul pružnosti se určí ze vztahu

$$E = \frac{F}{A_p \varepsilon}, \quad (2.1)$$



Obrázek 2.3: Linearizovaná materiálová charakteristika pásu

F [N]	5000	11 500	6000	2000	0
$\varepsilon$ [-]	0,02	0,075	0,062	0,04	0,018

Tabulka 2.2: Souřadnice materiálové charakteristiky polyesterového pásu

kde plocha průřezu  $A_p = b_p s_p$ . Rozměry pásu  $b_p$  a  $s_p$  jsou uvedeny dále. Hodnota Poissonova čísla je dle [18] zvolena  $\mu = 0,39$ .

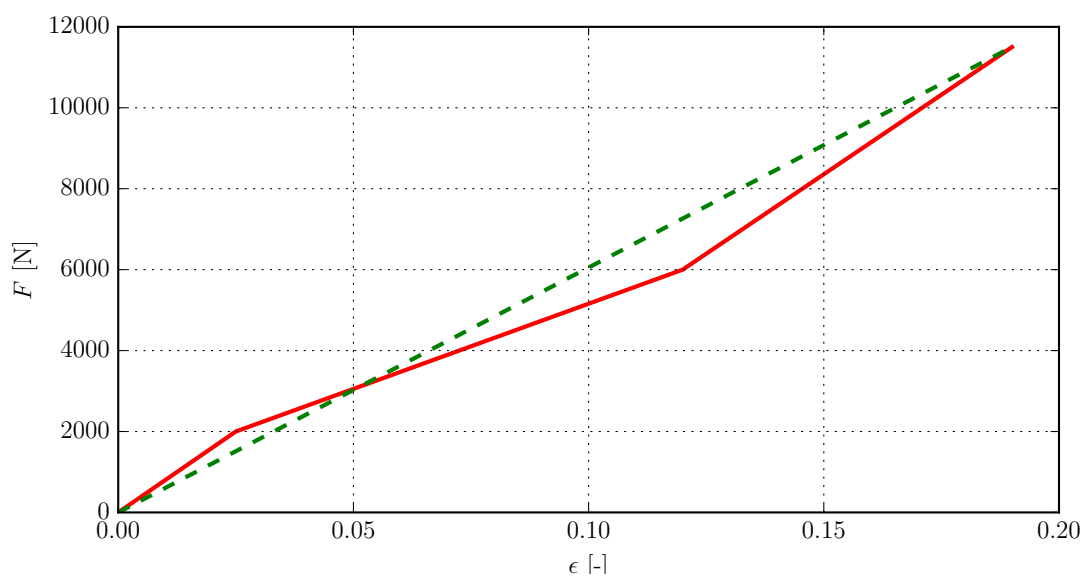
Výsledné materiálové konstanty, včetně hustoty  $\rho$ , jsou uvedeny v tab. 2.3.

$E$ [MPa]	$G$ [MPa]	$\mu$ [-]	$\rho$ [kg m <sup>-3</sup> ]
1050	375	0,39	1100

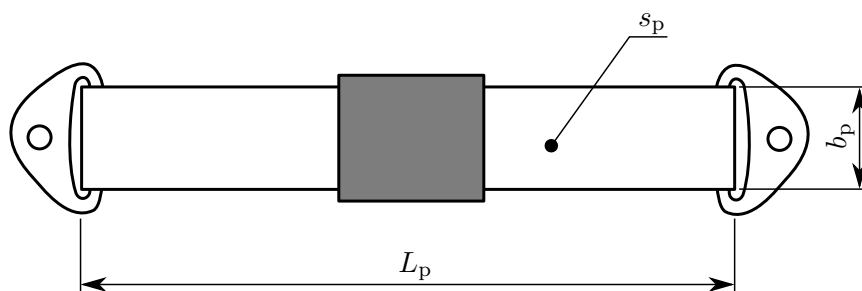
Tabulka 2.3: Materiálové konstanty

## 2.3 Geometrie bezpečnostního pásu

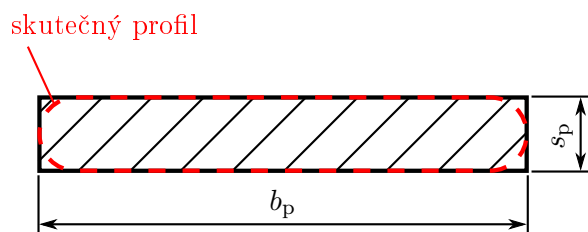
Geometrie bezpečnostního pásu se může mírně lišit v závislosti na výrobci a typu pásu. Šířka analyzovaného pásu je  $b_p = 48$  mm a tloušťka  $s_p = 1,2$  mm. Maximální délka pásu se pohybuje okolo  $L_p \approx 1400$  mm a je nastavitelná dle potřeby. Pás je možné prodloužit pomocí speciálního nástavce o maximální délce  $L_n \approx 630$  mm. Geometrie pásu je znázorněna na obr. 2.5 a jeho skutečný a idealizovaný průřez je znázorněn na obr. 2.6.



Obrázek 2.4: Linearizovaná materiálová křivka nylonového pásu



Obrázek 2.5: Rozměry bezpečnostního pásu



Obrázek 2.6: Profil průřezu bezpečnostního pásu



## 2.4 Bezpečnostní spony

Součástí pásu musí být kovová bezpečnostní blokovácí spona, která v případě nouze umožňuje rychlé rozepnutí pásu. Konstrukce spon se liší podle výrobce nebo požadavků zákazníka, funkčnost však musí zůstat zachována. Největší rozdíl je v konstrukci spony dvou, případně tříbodového pásu, a spony pro čtyř a pěti bodový pás. Příklad spony pro dvoubodový pás je na obr. 2.7.



Obrázek 2.7: Běžná spona bezpečnostního pásu [22],[23]

U tohoto typu dojde k rozepnutí pásu vyklopením horní části mechanismu spony. Konstrukce je vyrobena z vysokopevnostní hliníkové slitiny nebo z oceli a je dimenzována na zatížení v tahu  $F_S = 11 - 13 \text{ kN}$ . Stejný typ spony se používá rovněž pro tříbodový pás.

Alternativou je mechanismus podobný automobilovému zapínání pásu s tlačítkovým zámkem. Příklad je na obr. 2.8.



Obrázek 2.8: Tlačítková spona bezpečnostního pásu [24]

Byla provedena studie [8], která si kladla za cíl srovnat dobu potřebnou k rozepnutí pásu vybaveného běžnou vyklápěcí sponou nebo sponou tlačítkovou. Výsledky testování ukazují, že v průměru kratší čas potřebuje pasažér k rozepnutí klasické spony. Autoři však zároveň uvádějí, že potřebná doba je závislá na mnoha faktorech, jako jsou např. zkušenosti s daným mechanismem, věk, apod. Tento typ se používá v menší míře a to převážně v malých letadlech, případně ve vyšších cestovních třídách dopravních letadel.

U ostatních vícebodových pásů se používají rotační spony (obr. 2.9), u nichž dojde k rozepnutí pootočením mechanismu.



Obrázek 2.9: Rotační spona bezpečnostního pásu [25],[26]

Bezpečnostní pásy s rotační sponou se hojně využívají i v jiných odvětvích, jako je např. motorsport.

## Kapitola 3

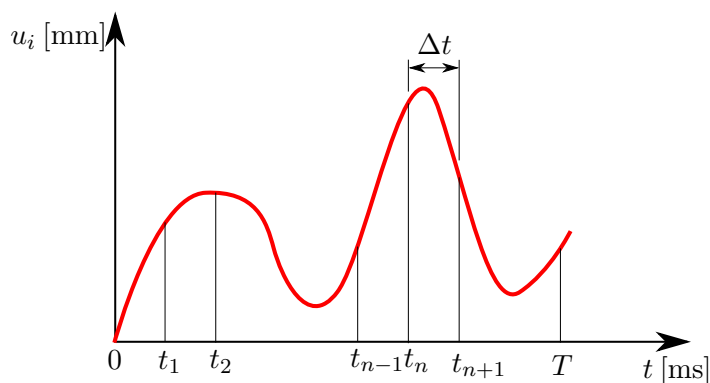
# Explicitní metoda řešení pohybových rovnic

V předchozích kapitolách byly stanoveny potřebné parametry testu a materiálové parametry bezpečnostního pásu. Dále bude odvozeno matematické pozadí řešení dynamických úloh a budou provedeny numerické simulace testu sedadla.

Pro řešení dynamických úloh pomocí metody konečných prvků je potřeba sestavit pohybovou rovnici zkoumané soustavy ve tvaru

$$\mathbf{M} \ddot{\mathbf{u}}(t) + \mathbf{C} \dot{\mathbf{u}}(t) + \mathbf{K} \mathbf{u}(t) = \mathbf{F}(t), \quad (3.1)$$

kde  $\mathbf{M}$  je matice hmotnosti,  $\mathbf{C}$  je matice tlumení,  $\mathbf{F}$  je vektor vnějších sil a  $\mathbf{u}$  je vektor posuvů. K řešení pohybové rovnice, resp. soustavy nelineárních obyčejných diferenciálních rovnic, je možno využít tzv. modální transformace, která převede soustavu na soustavu nezávislých rovnic. Další možností je použití diferenční metody, což vede k přímé integraci pohybových rovnic. Hledaným řešením je vektor posuvů  $\mathbf{u}(t)$ , případně jeho první a druhá derivace (vektor rychlosti a zrychlení). V závislosti na užitém diferenčním schématu se přímá integrace dělí na explicitní a implicitní. Oba způsoby řešení spočívají v nahrazení derivací vektoru posuvů diferencemi (diskretizaci spojitého průběhu hledané veličiny v čase), s časovým krokem  $\Delta t$  (obr. 3.1).



Obrázek 3.1: Příklad časového průběhu složky vektoru posuvů  $u_i$

Pohybová rovnice se převede na soustavu obecně nelineárních algebraických rovnic. Výběr integrační metody závisí na povaze řešeného problému a omezuje možnost volby velikosti časového kroku  $\Delta t$ . Pro numerickou stabilitu explicitní metody musí být  $\Delta t$  velmi malý a proto se s výhodou využívá k simulaci rychlých dějů, jako jsou např. crash testy, výbuchy, průstřely apod. Trvání těchto dějů se pohybuje v řádu milisekund a pro popis celé odezvy je nezbytné integrovat s dostatečně malým časovým krokem - ten se může navíc v průběhu výpočtu podle potřeby měnit. Naproti tomu implicitní metoda je za určitých předpokladů nepodmíněně stabilní a je možné volit  $\Delta t$  mnohem větší. Používá se k analýze pomalých nebo kvazistatických dějů, jako je např. tváření nebo obrábění. Kromě velikosti časového kroku se obě metody liší ve způsobu výpočtu posuvu v čase  $t + \Delta t$ . Zatímco u explicitního algoritmu je hledaná hodnota funkcí posuvu v čase  $t$  a  $t - \Delta t$ , u implicitního algoritmu navíc hledaný posuv závisí na dosud neznámé hodnotě posuvu v čase  $t + \Delta t$ . Dále bude podrobněji rozebrána explicitní metoda přímé integrace pohybové rovnice z důvodu její vhodnosti pro analyzování problematiky bezpečnostních pásů.

### 3.1 Odvození explicitní metody

Pro odvození explicitního algoritmu integrace pohybové rovnice v čase se využije aproximace výchylky v čase  $t + \Delta t$  a  $t - \Delta t$  pomocí Taylorova polynomu. Posuv v čase  $t + \Delta t$  pak lze vyjádřit jako

$$\mathbf{u}_{t+\Delta t} = \mathbf{u}_t + \Delta t \dot{\mathbf{u}}_t + \frac{\Delta t^2}{2} \ddot{\mathbf{u}}_t + \frac{\Delta t^3}{6} \dddot{\mathbf{u}}_t + \dots \quad (3.2)$$

a v čase  $t - \Delta t$

$$\mathbf{u}_{t-\Delta t} = \mathbf{u}_t - \Delta t \dot{\mathbf{u}}_t + \frac{\Delta t^2}{2} \ddot{\mathbf{u}}_t - \frac{\Delta t^3}{6} \dddot{\mathbf{u}}_t + \dots \quad (3.3)$$

Kubické a vyšší členy rozvoje lze za předpokladu rovnoměrně zrychleného pohybu pro malý časový krok zanedbat. Tím se do řešení vnese tzv. numerické tlumení. Následným odečtením 3.3 od 3.2 a úpravou se vyjádří vztah pro rychlost  $\dot{\mathbf{u}}$  v čase  $t$  ve tvaru

$$\dot{\mathbf{u}}_t = \frac{\mathbf{u}_{t+\Delta t} - \mathbf{u}_{t-\Delta t}}{2 \Delta t}. \quad (3.4)$$

Sečtením 3.2 a 3.3 se určí vztah pro zrychlení  $\ddot{\mathbf{u}}$  v čase  $t$  ve tvaru

$$\ddot{\mathbf{u}}_t = \frac{\mathbf{u}_{t-\Delta t} - 2 \mathbf{u}_t + \mathbf{u}_{t+\Delta t}}{\Delta t^2}. \quad (3.5)$$

Dosazením vztahů 3.4 a 3.5 do pohybové rovnice 3.1 se získá pohybová rovnice v diferenčním tvaru

$$\mathbf{M} \frac{\mathbf{u}_{t-\Delta t} - 2 \mathbf{u}_t + \mathbf{u}_{t+\Delta t}}{\Delta t^2} + \mathbf{C} \frac{\mathbf{u}_{t+\Delta t} - \mathbf{u}_{t-\Delta t}}{2 \Delta t} + \mathbf{K} \mathbf{u}_t = \mathbf{F}_t. \quad (3.6)$$

Po úpravě lze vyjádřit výchozí rovnici pro výpočet vektoru posuvu  $\mathbf{u}$  v čase  $t + \Delta t$

$$\left( \mathbf{M} + \frac{\Delta t \mathbf{C}}{2} \right) \mathbf{u}_{t+\Delta t} = \Delta t^2 (\mathbf{F}_t - \mathbf{K} \mathbf{u}_t) + \mathbf{M} (2 \mathbf{u}_t - \mathbf{u}_{t-\Delta t}) + \frac{\Delta t \mathbf{C}}{2} \mathbf{u}_{t-\Delta t}, \quad (3.7)$$

jak je uvedena např. v [12]. Je třeba si uvědomit, že matice hmotnosti  $\mathbf{M}$  a matice tlumení  $\mathbf{C}$  mohou mít všechny prvky obecně nenulové a při řešení takové rovnice nastává největší

problém při inverzi výsledné matice na levé straně rovnice  $\mathbf{M} + \frac{\Delta t \mathbf{C}}{2}$ . V praxi se proto často zavádí substituce ve tvaru

$$\frac{\mathbf{u}_{t+\Delta t} - \mathbf{u}_{t-\Delta t}}{2\Delta t} \approx \frac{\mathbf{u}_t - \mathbf{u}_{t-\Delta t}}{\Delta t}. \quad (3.8)$$

Dosazením substituce do rovnice 3.6 se po úpravě získá

$$\mathbf{M} \mathbf{u}_{t+\Delta t} = \Delta t^2 (\mathbf{F}_t - \mathbf{K} \mathbf{u}_t) + \mathbf{M} (2 \mathbf{u}_t - \mathbf{u}_{t-\Delta t}) + \Delta t \mathbf{C} (\mathbf{u}_{t-\Delta t} - \mathbf{u}_t), \quad (3.9)$$

kde na levé straně figuruje již pouze matice hmotnosti. Dalšího zjednodušení lze rovněž dosáhnout, bude-li uvažováno Rayleighovo tlumení. Pro přesné určení matice tlumení je potřeba provést náročné experimenty. Z tohoto důvodu se obvykle uvažuje viskózní model tlumení, který je závislý na pohybu. Matice tlumení se pak volí ve tvaru  $\mathbf{C} = \alpha \mathbf{M} + \beta \mathbf{K}$ , kde  $\alpha \mathbf{M}$  představuje vnější tlumení a  $\beta \mathbf{K}$  tlumení vnitřní (materiálové). Zanedbá-li se materiálové tlumení, bude  $\mathbf{C} = \alpha \mathbf{M}$ . Proveďte-li se nyní dosazení např. do rovnice 3.7 a vynásobí se celá rovnice inverzní maticí tuhosti  $\mathbf{M}^{-1}$ , zjednoduší se rovnice na tvar

$$\mathbf{u}_{t+\Delta t} = \frac{1}{2 + \alpha \Delta t} [2 \Delta t^2 \mathbf{M}^{-1} (\mathbf{F}_t - \mathbf{K} \mathbf{u}_t) + 4 \mathbf{u}_t + (\alpha \Delta t - 2) \mathbf{u}_{t-\Delta t}]. \quad (3.10)$$

V praxi se často používá diagonální matice hmotnosti, což má za následek výrazné zjednodušení výpočtu, neboť inverze diagonální matice je její převrácená hodnota. Pro výpočty bude uvažována nezjednodušená diskrétní rovnice 3.7.

### 3.1.1 Startovací procedura

Při podrobnějším pohledu na rovnici 3.7 je patrné, že v čase  $t = 0$ , tj. začátek výpočtu, je nutné znát výchylku v okamžiku před analyzovaným dějem, v čase  $t = -\Delta t$ . Pro zahájení výpočtu se používá tzv. startovací procedura, jejímž úkolem je nalézt neznámou výchylku. Využije se zde opět Taylorova rozvoje

$$\mathbf{u}_{t-\Delta t} = \mathbf{u}_t - \Delta t \dot{\mathbf{u}}_t + \frac{\Delta t^2}{2} \ddot{\mathbf{u}}_t. \quad (3.11)$$

Dosazením počátečních podmínek v čase  $t = 0$ :  $\mathbf{u} = \mathbf{u}_0$ ,  $\dot{\mathbf{u}} = \dot{\mathbf{u}}_0$  se získá rovnice ve tvaru

$$\mathbf{u}_{-\Delta t} = \mathbf{u}_0 - \Delta t \dot{\mathbf{u}}_0 + \frac{\Delta t^2}{2} \ddot{\mathbf{u}}_0. \quad (3.12)$$

Neznámé  $\ddot{\mathbf{u}}_0$  se určí z pohybové rovnice 3.1 dosazením počátečních podmínek

$$\mathbf{M} \ddot{\mathbf{u}}_0 + \mathbf{C} \dot{\mathbf{u}}_0 + \mathbf{K} \mathbf{u}_0 = \mathbf{F}_0, \quad (3.13)$$

jako

$$\ddot{\mathbf{u}}_0 = \mathbf{M}^{-1} (\mathbf{F}_0 - \mathbf{C} \dot{\mathbf{u}}_0 - \mathbf{K} \mathbf{u}_0). \quad (3.14)$$

### 3.1.2 Časový krok

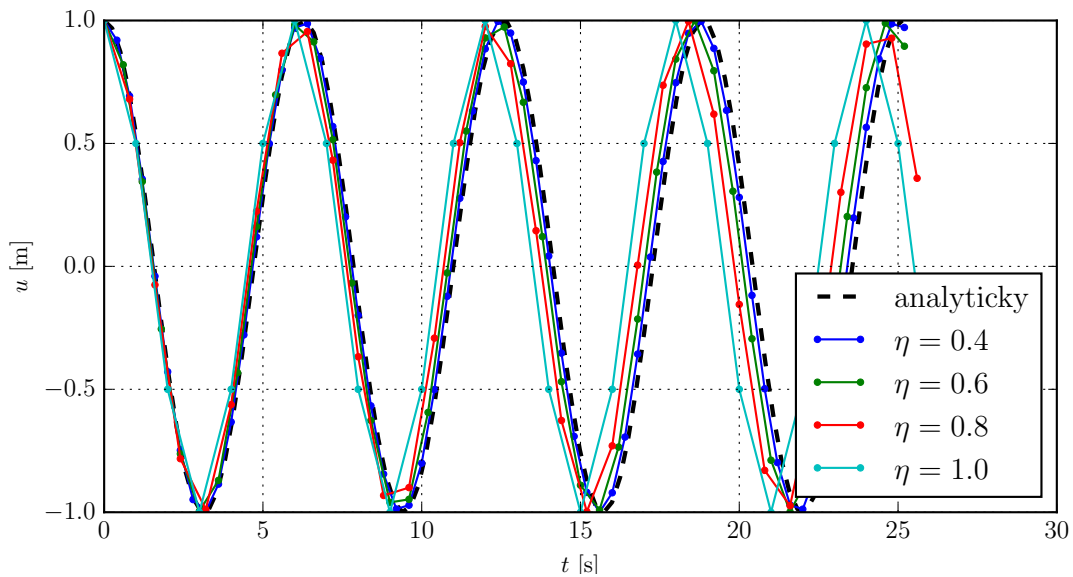
Volba časového kroku  $\Delta t$  je u explicitního algoritmu značně limitována a na jeho volbě závisí, zda bude výpočet reálně popisovat skutečnost či nikoliv. Rovnice 3.7, 3.9 a 3.10 jsou podmíněně stabilní a dle [6] musí platit, že

$$\Delta t \leq \frac{2}{\Omega_{\max}} = \frac{T_{\min.}}{\pi} = \Delta t_{\text{crit.}}. \quad (3.15)$$

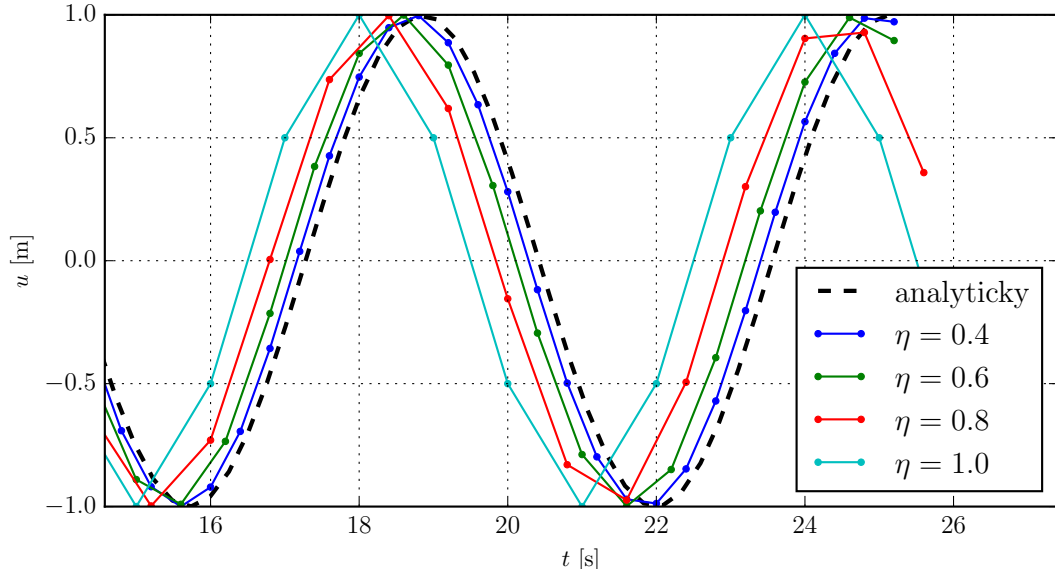
$\Omega_{\max}$ , resp.  $T_{\min}$  závisí na parametrech sítě konečných prvků a je to maximální hodnota vlastní kruhové frekvence, resp. nejmenší hodnota periody vlastního netlumeného kmitání ze všech prvků. Velikost časového kroku je možno dále snížit pomocí faktoru  $\eta$

$$\Delta t = \eta \Delta t_{\text{crit.}}, \quad (3.16)$$

kde faktor  $\eta \in (0, 1)$ . Vliv velikosti faktoru  $\eta$  je demonstrován na příkladu kmitání diskretního systému s 1 stupněm volnosti (příklad v kapitole 3.2.1a) a průběh výchylky při různé hodnotě  $\eta$  je patrný z obr. 3.2 a 3.3. **S klesající velikostí časového kroku se numerické řešení blíží analytickému, zároveň tím však roste celkový čas potřebný pro výpočet.**



Obrázek 3.2: Vliv velikosti časového kroku



Obrázek 3.3: Vliv velikosti časového kroku - detail

## 3.2 Aplikace explicitní metody

Jedním z hlavních cílů této práce je **vytvoření výpočetního algoritmu v programovacím jazyce Python pro zjednodušenou analýzu bezpečnostního pásu**. Algoritmus je postupně odladěn na několika příkladech výpočtu dynamického chování diskrétního i spojitého systému, u nichž je známo analytické řešení. Úlohy jsou řazeny od elementárních - diskrétní systém s 1 stupněm volnosti - až po komplexní řešení systému kontinua pomocí MKP, se zahrnutím vlivu geometrických nelinearit. Každá úloha určitým způsobem rozšiřuje úlohu předcházející a vlastní algoritmus. Všechny příklady jsou uvedeny dále v kapitolách 3.2.1a až 3.2.2c. V kapitole 4.1 jsou pak využity všechny poznatky pro samotnou analýzu pásu.

### Úprava rovnice pro implementaci

Pro implementaci algoritmu je výhodné rovnici 3.7 formálně upravit na tvar

$$”\mathbf{A} \mathbf{x} = \mathbf{b}”, \quad (3.17)$$

kde  $\mathbf{A}$  je matice koeficientů,  $\mathbf{x}$  je vektor neznámých a  $\mathbf{b}$  je vektor pravé strany. Rovnice, resp. soustava rovnic v tomto tvaru je pak řešitelná různými metodami ať už přímými, jako např. Choleského rozklad, LU rozklad, nebo iteračními, mezi které patří např. metoda sdružených gradientů. Samotná úprava rovnice pak spočívá v substituci členů původní rovnice

$$\left( \mathbf{M} + \frac{\Delta t \mathbf{C}}{2} \right) \mathbf{u}_{t+\Delta t} = \Delta t^2 (\mathbf{F}_t - \mathbf{K} \mathbf{u}_t) + \mathbf{M} (2 \mathbf{u}_t - \mathbf{u}_{t-\Delta t}) + \frac{\Delta t \mathbf{C}}{2} \mathbf{u}_{t-\Delta t}$$

na hledaný tvar

$$\mathbf{M}_{\text{eff}} \mathbf{u}_{t+\Delta t} = \mathbf{F}_{\text{eff}}, \quad (3.18)$$

kde matice koeficientů je označena jako  $\mathbf{M}_{\text{eff}}$ , vektor pravé strany jako  $\mathbf{F}_{\text{eff}}$  a vektorem neznámých je zde vektor posuvů v čase  $t + \Delta t$   $\mathbf{u}_{t+\Delta t}$ .

### Obecné schéma explicitní metody

Algoritmus samotného výpočtu lze obecně rozvrhnout do následujících kroků:

1. Specifikace materiálových dat
2. Načtení sítě konečných prvků
3. Specifikace parametrů algoritmu (velikost kroku  $\Delta t$ , celkový čas)
4. Specifikace počátečních podmínek  $\mathbf{u}_0, \dot{\mathbf{u}}_0, \mathbf{F}_0$
5. Sestavení globálních matic soustavy  $\mathbf{K}, \mathbf{M}$  na základě transformace
6. Specifikace vektoru zatížení  $\mathbf{F}_t$
7. Startovací procedura - výpočet  $\mathbf{u}_{-\Delta t}$
8. Začátek iterační smyčky:
  - 8.1 Sestavení vektoru pravé strany  $\mathbf{F}_{\text{eff}}$
  - 8.2 Sestavení matice koeficientů  $\mathbf{M}_{\text{eff}}$
  - 8.3 Řešení soustavy  $\mathbf{M}_{\text{eff}} \mathbf{u}_{t+\Delta t} = \mathbf{F}_{\text{eff}}$
  - 8.4 Uložení výsledků
  - 8.5 Transformace matice tuhosti podle nových souřadnic uzlů

Konec iterační smyčky

9. Konec výpočtu

Transformace uvedená u bodu 5 je počáteční transformace matic daná geometrií úlohy. Transformace u bodu 8.5 se pro malé posuvy a deformace neprojeví. V takovém případě lze tento bod přeskočit. Všechny případy budou demonstrovány na příkladech. Algoritmus je možno dále rozšířit např. o automatickou změnu časového kroku v závislosti na změně průřezových charakteristik konečných prvků v důsledku jejich deformace. Později, v kapitole 3.2.2, budou do algoritmu zahrnuty geometrické nelinearity ve formě velkých posunutí, resp. natočení elementů.



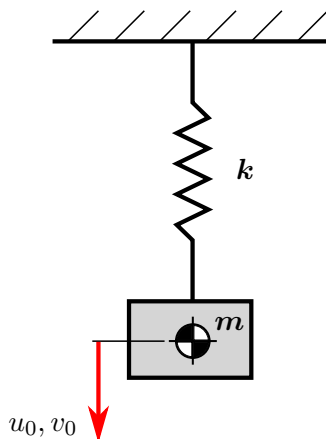
### 3.2.1 Diskrétní systém

Prvotní ověření explicitního algoritmu je provedeno analýzou diskrétního systému typu hmotný bod - pružina na příkladu volného netlumeného a tlumeného kmitání. Aplikace na obdobném příkladu je uvedena v [6].

#### 3.2.1a Příklad volného netlumeného kmitání s 1 stupněm volnosti

##### Popis úlohy

Cílem analýzy je určit časový průběh výchylky  $\mathbf{u}(t)$  diskrétního systému tvořeného 1 tělesem - hmotným bodem o hmotnosti  $m = 1 \text{ kg}$ . Těleso kmitá na lineární pružině o tuhosti  $k = 1 \text{ Nm}^{-1}$ . Počáteční výchylka v čase  $t = 0$  je  $u_0 = 1 \text{ m}$  a počáteční rychlost  $\dot{u}_0 = 0$ . Tato úloha odpovídá řešení kmitání předpruženého tělesa po jeho uvolnění. Analytické řešení je odvozeno např v [11]. Schéma úlohy je na obr. 3.4.



Obrázek 3.4: Schéma úlohy netlumeného kmitání s 1 stupněm volnosti

##### Analytické řešení

Netlumené kmitání soustavy s 1 stupněm volnosti je možné popsat pohybovou rovnicí

$$m \ddot{u} + k u = 0, \quad (3.19)$$

jejíž řešení lze odvodit ve tvaru

$$u(t) = u_a \sin(\Omega_0 t + \varphi_0), \quad (3.20)$$

kde vlastní kruhová frekvence netlumeného kmitání je

$$\Omega_0 = \sqrt{\frac{k}{m}} \quad (3.21)$$

a  $u_a$  a  $\varphi_0$  jsou integrační konstanty, které představují amplitudu kmitání a fázový posuv a určí se z počátečních podmínek v čase  $t = 0$ :  $u = u_0$ ,  $\dot{u} = \dot{u}_0$  jako

$$u_a = \sqrt{u_0^2 + \frac{\dot{u}_0^2}{\Omega_0^2}} \quad (3.22)$$

a

$$\varphi_0 = \arctan\left(\frac{u_0 \Omega_0}{\dot{u}_0}\right). \quad (3.23)$$

Řešením rovnice 3.20 pro dané počáteční podmínky  $u_0$  a  $\dot{u}_0$  se získá průběh výchylky tělesa v čase  $t$ .

### Numerické řešení explicitní metodou

Pro numerické řešení explicitní metodou je zapotřebí převést pohybovou rovnici 3.19 na tvar 3.7, resp. 3.18. V případě **netlumeného** systému s **1 stupněm volnosti** přejde maticová rovnice 3.7 na skalární rovnici ve tvaru

$$m u_{t+\Delta t} = \Delta t^2 (F_t - k u_t) + m (2 u_t - u_{t-\Delta t}). \quad (3.24)$$

Startovací procedura bude mít tvar

$$u_{-\Delta t} = u_0 - \Delta t \dot{u}_0 + \frac{\Delta t^2}{2} \ddot{u}_0, \quad (3.25)$$

kde

$$\ddot{u}_0 = \frac{1}{m} (F_0 - k u_0). \quad (3.26)$$

Pro volné kmitání se rovnice 3.24 ještě zjednoduší na

$$m u_{t+\Delta t} = -\Delta t^2 k u_t + m (2 u_t - u_{t-\Delta t}), \quad (3.27)$$

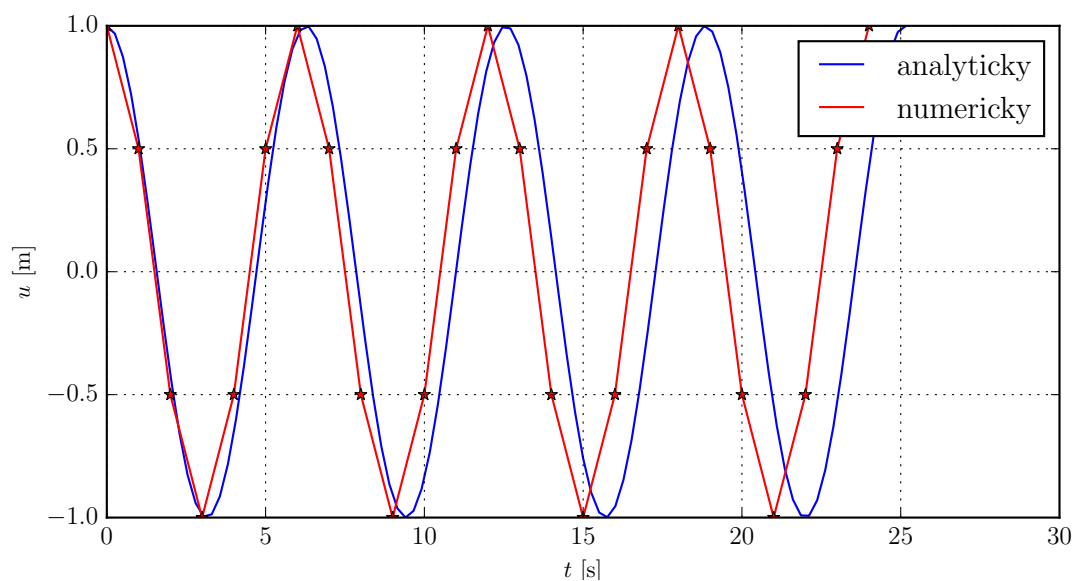
a rovnice 3.26 na

$$\ddot{u}_0 = -\frac{k}{m} u_0 = -\Omega_0^2 u_0. \quad (3.28)$$

Úpravou rovnice 3.27 se získá hledané numerické řešení pohybové rovnice 3.19 v diferenčním tvaru

$$u_{t+\Delta t} = -\Delta t^2 \Omega_0^2 u_t + 2 u_t - u_{t-\Delta t}. \quad (3.29)$$

Řešením rovnice 3.29 pro dané počáteční podmínky  $u_0$  a  $\dot{u}_0$  a časový krok  $\Delta t$  se získá opět průběh výchylky tělesa v čase  $t$ . Numerické řešení pohybové rovnice je pro jednotnost provedeno s časovým krokem  $\Delta t = 1$  s, který odpovídá podmínce 3.15. Srovnání časového průběhu výchylky získaného analyticky a numericky je na obr. 3.5. Je patrné, že pro dané parametry systému a zvolený časový krok se odezva vypočtená explicitním algoritmem mírně zpožďuje za analytickým řešením. Porovnání průběhů pro jiné velikosti časového kroku je uvedeno již dříve na obr. 3.2 a 3.3.

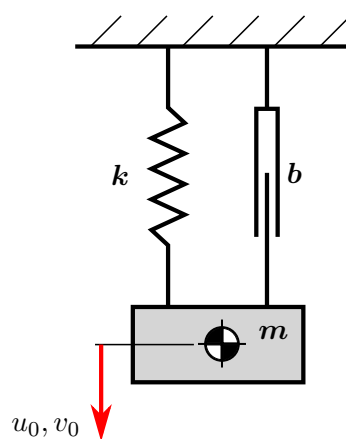


Obrázek 3.5: Analytické a numerické řešení netlumeného kmitání

### 3.2.1b Příklad volného tlumeného kmitání s 1 stupněm volnosti

#### Popis úlohy

Pro ověření vlivu tlumení v explicitních rovnicích je zde uveden příklad volného tlumeného kmitání, který je k nalezení rovněž v [11]. Zadání úlohy je stejné jako pro volné kmitání v předešlé kapitole 3.2.1a. Navíc zde však přibude tlumení s koeficientem  $b = 0,1 \text{ Nm}^{-1}\text{s}$ . Úloha představuje např. kmitání předpruženého tělesa v kapalině a její schéma je na obr. 3.6.



Obrázek 3.6: Schéma úlohy tlumeného kmitání s 1 stupněm volnosti

## Analytické řešení

Tlumené kmitání soustavy s 1 stupněm volnosti lze popsat pohybovou rovnicí

$$m \ddot{u} + b \dot{u} + k u = 0, \quad (3.30)$$

jejíž řešení se dá odvodit ve tvaru

$$u(t) = u_a e^{-\delta t} \sin(\Omega t + \varphi_0). \quad (3.31)$$

Konstanta doznívání je definována jako

$$\delta = \frac{b}{2m} \quad (3.32)$$

a vlastní kruhová frekvence

$$\Omega = \sqrt{\Omega_0^2 - \delta^2}. \quad (3.33)$$

Integrační konstanty  $u_a$  a  $\varphi_0$  představují amplitudu kmitání a fázový posuv a určí se z počátečních podmínek v čase  $t = 0$ :  $u = u_0$ ,  $\dot{u} = \dot{u}_0$  ve tvaru

$$u_a = \sqrt{u_0^2 + \frac{(\dot{u}_0 + u_0 \delta)^2}{\Omega^2}} \quad (3.34)$$

a

$$\varphi_0 = \arctan\left(\frac{u_0 \Omega}{\dot{u}_0 + u_0 \delta}\right). \quad (3.35)$$

Řešením rovnice 3.31 pro dané počáteční podmínky  $u_0$  a  $\dot{u}_0$  se získá průběh výchylky tělesa v čase  $t$ .

## Numerické řešení explicitní metodou

Numerické řešení kmitání tlumeného systému je analogické s řešením systému netlumeného. Rovnice však také obsahuje členy, které zohledňují tlumení. V případě **tlumeného** systému s **1 stupněm volnosti** tedy přejde maticová rovnice 3.7 na skalární rovnici ve tvaru

$$\left(m + \frac{\Delta t b}{2}\right) u_{t+\Delta t} = \Delta t^2 (F_t - k u_t) + m (2 u_t - u_{t-\Delta t}) + \frac{\Delta t b}{2} u_{t-\Delta t}. \quad (3.36)$$

Startovací procedura má tvar

$$u_{-\Delta t} = u_0 - \Delta t \dot{u}_0 + \frac{\Delta t^2}{2} \ddot{u}_0, \quad (3.37)$$

kde

$$\ddot{u}_0 = \frac{1}{m} (F_0 - b \dot{u}_0 - k u_0). \quad (3.38)$$

Pro volné kmitání se rovnice 3.36 ještě zjednoduší na

$$\left(m + \frac{\Delta t b}{2}\right) u_{t+\Delta t} = -\Delta t^2 k u_t + m (2 u_t - u_{t-\Delta t}) + \frac{\Delta t b}{2} u_{t-\Delta t} \quad (3.39)$$

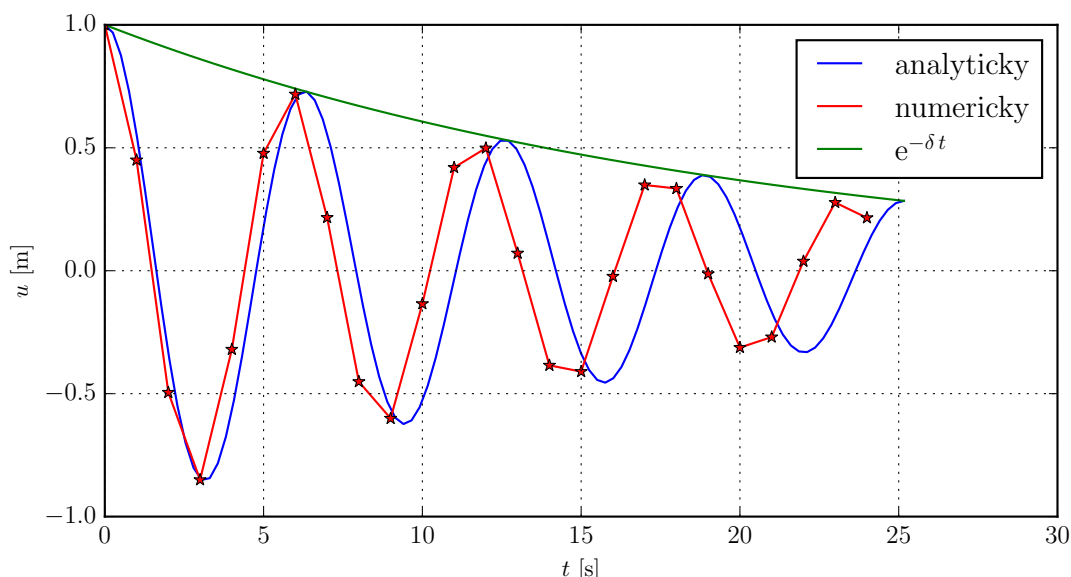
a 3.38 na

$$\ddot{u}_0 = -\frac{b}{m} \dot{u}_0 - \frac{k}{m} u_0 = -2\delta \dot{u}_0 - \Omega^2 u_0. \quad (3.40)$$

Úpravou 3.39 se získá hledané numerické řešení pohybové rovnice 3.30 v diferenčním tvaru

$$u_{t+\Delta t} = \frac{1}{1 + \Delta t \delta} \left( -\Delta t^2 \Omega_0^2 u_t + 2u_t - u_{t-\Delta t} + \Delta t \delta u_{t-\Delta t} \right). \quad (3.41)$$

Numerické řešení je provedeno pouze pro časový krok o velikosti  $\Delta t = 1$  s. Srovnání časového průběhu výchylky získaného analyticky a numericky je na obr. 3.7. Pro dané parametry je opět patrné zpoždění odezvy získané numericky oproti analytickému řešení. V grafu je navíc vykreslena obálka funkce analytického řešení, která je popsána členem  $e^{-\delta t}$  v rovnici 3.30 a má exponenciální průběh.



Obrázek 3.7: Analytické a numerické řešení tlumeného kmitání

Skript použitý pro výpočet předešlých úloh v Pythonu je uveden v příloze A. Jedná se o stejný algoritmus pro tlumený i netlumený systém. Volba se provádí pouze zadáním hodnoty součinitele  $b$ . Tedy pro netlumený systém  $b = 0$  a pro tlumený  $b > 0$ . V kódu je zahrnuto i vykreslení průběhů pro proměnnou velikost časového kroku z obr. 3.2 a 3.3.

Na těchto dvou příkladech byl ověřen především vliv velikosti časového kroku a správnost odvozeného explicitního algoritmu se zahrnutím vlivu vnějšího tlumení.

### 3.2.2 Kontinuum - maticová formulace

Dosud uvedené příklady byly řešeny bez využití MKP pouze pro ověření funkčnosti algoritmu. Pro další úlohy již bude zavedena konečnoprvková formulace řešených problémů. Ve všech následujících aplikacích bude použit dvouuzlový tyčový prvek.

#### Nelineární schéma explicitní metody

V předchozích příkladech byla uvažována konstantní tuhost elementů v celém časovém intervalu výpočtu. U mnoha reálných případů je však tuhost funkcí času, resp. posunutí uzlů. To nastává při zahrnutí nelinearity do výpočtu, které lze např. dle [2] rozdělit do několika hlavních skupin.

Prvním typem jsou materiálové nelinearity, u kterých se předpokládá zanedbatelné posunutí uzlů a deformace elementů, ale napětově-deformační vztahy jsou nelineární. Tyto nelinearity je nutné zahrnout do výpočtu při použití materiálu s nelineárním napětově-deformačním chováním, jako např. pryž, nebo při vzniku plastických deformací v součásti. Závisí tedy na použitém materiálu, resp. na volbě materiálového modelu. Pro výpočet napětí a deformací se zde používají vztahy pro tzv. inženýrské napětí a deformaci. **V této práci bude pro výpočet uvažován materiál s lineárním chováním v elastické oblasti, tedy bude předpokládána platnost Hookeova zákona.**

Dalším typem nelinearity jsou nelinearity geometrické. Ty jsou zapříčiněny vždy velkými posuvy, příp. rotacemi elementů, přičemž v prvním případě dochází i k velkým deformacím prvků, zatímco ve druhém případě jsou deformace malé. V obou případech mohou být napětově-deformační vztahy lineární nebo nelineární. Dochází-li k velkým posuvům, je nutné počítat s veličinami v posunutých polohách. Pro popis pohybu kontinua vzhledem k referenčnímu souřadnému systému se ve strukturální mechanice používá Lagrangeova formulace, která sleduje pohyb bodů kontinua. Ta se dále dělí na totální Lagrangeovu formulaci, kdy jsou všechny veličiny vztaženy k původní konfiguraci a na aktualizovanou Lagrangeovu formulaci, u které jsou veličiny vztaženy ke konfiguraci na začátku každého zátěžného kroku. Pro výpočet napětí a deformací se zahrnutím geometrických nelinearit se nejčastěji používá Green-Lagrangeův tenzor přetvoření ve spojení s 2. Piolla-Kirchhoffovým tenzorem napětí. Výhodou těchto tenzorů je, že jsou invariantní vůči posuvům tělesa jako tuhého celku. 2. Piolla-Kirchhoffův však nemá přímý fyzikální význam a pro vyhodnocení napjatosti je nutné jej přepočítat pomocí převodních vztahů na Cauchyho tenzor napětí, který odpovídá skutečnému napětí v součásti. **Tento druh nelinearit je zahrnut do následujících aplikací.**

Pro úplnost je zde uveden ještě jiný používaný způsob popisu pohybu kontinua, kterým je Eulerova formulace. Ta se používá převážně pro analýzy v mechanice tekutin a pro výpočet napětí a přetvoření se používá Almasiho tenzor přetvoření ve spojení s již zmíněným Cauchyho tenzorem napětí.

Posledním typem nelinearit jsou kontakty, případně nelinearity způsobené změnou okrajových podmínek. **Tyto nebudou ve vlastní aplikaci uvažovány.**

### Rovinný tyčový prvek

V následujících kapitolách bude pro další ladění algoritmu a jeho následnou aplikaci u zjednodušené analýzy bezpečnostního pásu využít **rovinný dvouuzlový tyčový prvek**, se dvěma stupni volnosti v každém uzlu. Pro tento element budou dále stručně odvozeny příslušné matice hmotnosti a tuhosti.

### Matice hmotnosti

Odvození vychází z jednoho ze základních principů MKP - z aproximace pole posunutí po částech spojitou funkcí, viz [7]. Pro tyčový prvek v rovině lze vyjádřit pole posuvů nad elementem aproximační rovnicí

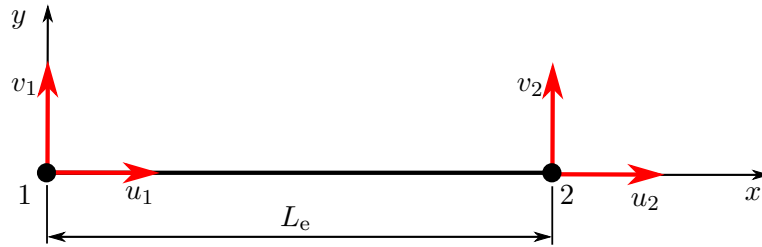
$$\mathbf{u} = \mathbf{N}\tilde{\mathbf{u}}, \quad (3.42)$$

kde  $\mathbf{N}$  je matice tvarových funkcí, která má pro aproximaci posuvů polynomem prvního stupně tvar

$$\mathbf{N} = \begin{bmatrix} 1 - \frac{x}{L_e} & 0 & \frac{x}{L_e} & 0 \\ 0 & 1 - \frac{x}{L_e} & 0 & \frac{x}{L_e} \end{bmatrix} \quad (3.43)$$

a  $\tilde{\mathbf{u}}$  je vektor zobecněných uzlových posunutí ve tvaru

$$\tilde{\mathbf{u}} = \left\{ \begin{matrix} u_1 & v_1 & u_2 & v_2 \end{matrix} \right\}^T. \quad (3.44)$$



Obrázek 3.8: Stupně volnosti elementu v lokálním souřadném systému

Po roznásobení se získají aproximační rovnice posuvů ve dvou směrech

$$\mathbf{u} = \begin{Bmatrix} u \\ v \end{Bmatrix} = \begin{Bmatrix} u_1 \left(1 - \frac{x}{L_e}\right) + u_2 \frac{x}{L_e} \\ v_1 \left(1 - \frac{x}{L_e}\right) + v_2 \frac{x}{L_e} \end{Bmatrix}. \quad (3.45)$$

Matice hmotnosti pak lze získat např. ze vztahu pro kinetickou energii elementu, která je definována jako

$$\Pi_k = \frac{1}{2} \int_V \dot{\mathbf{u}}^2 \rho dV = \frac{1}{2} \dot{\mathbf{u}}^T \int_V \mathbf{N}^T \mathbf{N} \rho dV \dot{\mathbf{u}}. \quad (3.46)$$

Přepsáním rovnice pro jednodimenzionální prvek s konstantním průřezem po celé délce pak lze psát

$$\Pi_k = \frac{1}{2} \dot{\mathbf{u}}^\top \rho A \int_{L_e} \mathbf{N}^\top \mathbf{N} dx \dot{\mathbf{u}}. \quad (3.47)$$

Z předchozí rovnice 3.47 plyne vztah pro výpočet matice hmotnosti prvku

$$\mathbf{M} = \rho A \int_0^{L_e} \mathbf{N}^\top \mathbf{N} dx. \quad (3.48)$$

Dosazením z 3.43 a integrací se získá tzv. konzistentní matice hmotnosti elementu ve tvaru

$$\mathbf{M} = \frac{\rho A L_e}{6} \begin{bmatrix} 2 & 0 & 1 & 0 \\ 0 & 2 & 0 & 1 \\ 1 & 0 & 2 & 0 \\ 0 & 1 & 0 & 2 \end{bmatrix}. \quad (3.49)$$

V dynamických analýzách, zvláště pak v analýzách řešených explicitními metodami, se však konzistentní matice zpravidla nepoužívá. Z rovnice 3.2 je zřejmé, že pro výpočet neznámé výchylky v čase  $t + \Delta t$  je potřeba provést inverzi matice hmotnosti. Tato operace je značně náročná a roste s počtem stupňů volnosti soustavy. Řešení výsledné soustavy rovnic se pak prodlužuje. Explicitní algoritmus se stává **efektivní při použití diagonální matice hmotnosti**, jejíž inverze je velice snadná a rychlá. Tím se řešený vázaný problém o  $n$  stupních volnosti převede na soustavu  $n$  lineárně nezávislých rovnic o  $n$  neznámých. Ty pak lze řešit každou samostatně, čímž se značně urychlí výpočet.

Konzistentní matici lze diagonalizovat např. přičtením mimodiagonálních členů v každém řádku nebo sloupci na hlavní diagonálu, jak uvádí např. [13]. Touto úpravou se pak získá matice hmotnosti ve tvaru

$$\mathbf{M} = \frac{\rho A L_e}{2} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3.50)$$

Tato matice popisuje rovnoměrné rozdělení hmotnosti do uzlů, kde každému uzlu je přiřazena polovina hmotnosti elementu.

## Matice tuhosti

Pro odvození matice tuhosti je dle [7] využit vztah pro deformační energii elementu, který má tvar

$$\Pi_d = \frac{1}{2} \int_V \epsilon^\top \sigma dV. \quad (3.51)$$



Využitím vektorové notace lze psát vektor poměrných deformací

$$\boldsymbol{\epsilon} = \left\{ \begin{matrix} \epsilon_{11} & \epsilon_{22} & \epsilon_{33} & \gamma_{12} & \gamma_{23} & \gamma_{13} \end{matrix} \right\}^T \quad (3.52)$$

a vektor napětí

$$\boldsymbol{\sigma} = \left\{ \begin{matrix} \sigma_{11} & \sigma_{22} & \sigma_{33} & \tau_{12} & \tau_{23} & \tau_{13} \end{matrix} \right\}^T. \quad (3.53)$$

Vektor napětí lze dále přepsat ve formě Hookeova zákona za pomoci matice materiálových konstant  $\mathbf{D}$  ve tvaru

$$\boldsymbol{\sigma} = \mathbf{D} \boldsymbol{\epsilon} \quad (3.54)$$

Při jednoosém namáhání vzniká v tyčovém prvku jediné nenulové napětí, a to v ose elementu. Zde toto napětí odpovídá  $\sigma_{11} = \sigma_x$ . Odpovídající přetvoření je pak  $\epsilon_{11} = \epsilon_x$ . Vztah pro poměrnou deformaci v ose elementu se získá z Green-Lagrangeova tenzoru velkých přetvoření ve tvaru

$$\epsilon_x = \frac{\partial u}{\partial x} + \frac{1}{2} \left[ \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial x} \right)^2 \right]. \quad (3.55)$$

Mnoho autorů zanedbává první člen  $\left( \frac{\partial u}{\partial x} \right)^2$  (např. [6] nebo [3]). V této práci bude tento člen uvažován, což kromě zvýšení přesnosti výpočtu s sebou přináší i výhodu při transformaci matic, jak je ukázáno dále. Vektor napětí 3.54 se pak zredukuje na tvar prostého Hookeova zákona

$$\sigma_x = E \epsilon_x, \quad (3.56)$$

kde  $E$  je modul pružnosti v tahu. Dosazením posledních dvou vztahů do rovnice 3.51 a uvážením jednorozměrného prvku s konstantním průřezem po celé jeho délce se získá vztah pro deformační energii tyčového elementu v integrálním tvaru

$$\Pi_d = \frac{EA}{2} \int_0^{L_e} \left[ \frac{\partial u}{\partial x} + \frac{1}{2} \left[ \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial x} \right)^2 \right] \right]^2 dx. \quad (3.57)$$

Po roznásobení má rovnice tvar

$$\Pi_d = \frac{EA}{2} \int_0^{L_e} \left[ \left( \frac{\partial u}{\partial x} \right)^2 + \left( \frac{\partial u}{\partial x} \right)^3 + \frac{\partial u}{\partial x} \left( \frac{\partial v}{\partial x} \right)^2 + \frac{1}{4} \left[ \left( \frac{\partial u}{\partial x} \right)^4 + 2 \left( \frac{\partial u}{\partial x} \right)^2 \left( \frac{\partial v}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial x} \right)^4 \right] \right] dx. \quad (3.58)$$

Poslední člen integrálu  $\frac{1}{4} \left[ \left( \frac{\partial u}{\partial x} \right)^4 + 2 \left( \frac{\partial u}{\partial x} \right)^2 \left( \frac{\partial v}{\partial x} \right)^2 + \left( \frac{\partial v}{\partial x} \right)^4 \right]$  se z důvodu linearizace často zanedbává a ani zde není dále uvažován. Po dosazení složek posuvů z rovnice 3.45 a provedení příslušných operací se získá výsledný vztah pro deformační energii ve tvaru

$$\Pi_d = \frac{1}{2} \tilde{\mathbf{u}}^T \frac{EA}{L_e} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} \tilde{\mathbf{u}} + \frac{1}{2} \tilde{\mathbf{u}}^T \frac{EA(u_2 - u_1)}{L_e^2} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix} \tilde{\mathbf{u}}, \quad (3.59)$$

nebo pomocí maticového zápisu

$$\Pi_d = \frac{1}{2} \tilde{\mathbf{u}}^T \mathbf{K}_T \tilde{\mathbf{u}} + \frac{1}{2} \tilde{\mathbf{u}}^T \mathbf{K}_G \tilde{\mathbf{u}}, \quad (3.60)$$

kde  $\mathbf{K}_T$  je elastická matice tuhosti a  $\mathbf{K}_G$  je matice geometrické tuhosti. Výsledná matice tuhosti tyčového elementu pro velké deformace pak vznikne součtem elastické a geometrické matice tuhosti jako

$$\mathbf{K}_L = \mathbf{K}_T + \mathbf{K}_G = \frac{EA}{L_e} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 0 & 0 & 0 \\ -1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + \frac{EA(u_2 - u_1)}{L_e^2} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}. \quad (3.61)$$

Člen  $\frac{EA(u_2 - u_1)}{L_e^2}$  představuje závislost na osově síle v každém časovém kroku a lze psát

$$\frac{EA(u_2 - u_1)}{L_e^2} = \frac{N}{L_e} \quad (3.62)$$

### Transformace do globálních souřadnic

Dosud bylo odvození provedeno na úrovni lokálních souřadnic elementu. Dochází-li v úloze k natočení elementu oproti výchozí vodorovné poloze, je nutné odvozené vztahy pro převedení do globálního souřadného systému transformovat pomocí transformační matice, která má dle [2] pro rovinný tyčový pvek tvar

$$\mathbf{T} = \begin{bmatrix} \cos(\varphi) & \sin(\varphi) & 0 & 0 \\ -\sin(\varphi) & \cos(\varphi) & 0 & 0 \\ 0 & 0 & \cos(\varphi) & \sin(\varphi) \\ 0 & 0 & -\sin(\varphi) & \cos(\varphi) \end{bmatrix}. \quad (3.63)$$

Natočení elementu je znázorněno na obr. 3.9.

Úhel natočení  $\varphi$ , resp.  $\cos(\varphi)$  a  $\sin(\varphi)$  lze s výhodou určit např. pomocí skalárního a vektorového součinu dvou vektorů

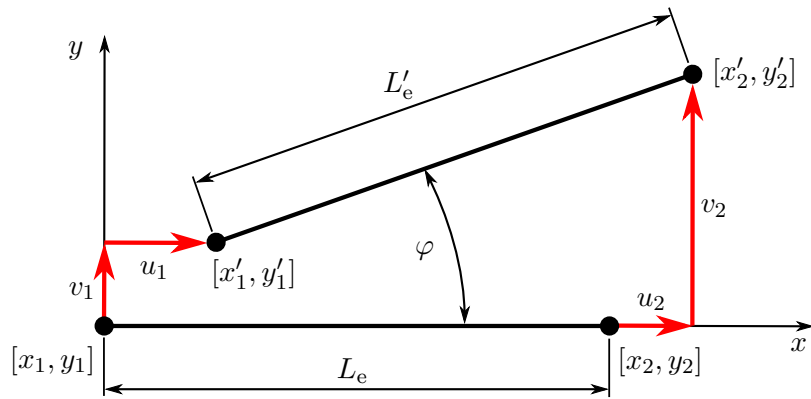
$$\cos(\varphi) = \frac{\mathbf{l} \cdot \mathbf{l}'}{|\mathbf{l}| |\mathbf{l}'|} \quad (3.64)$$

a

$$\sin(\varphi) = \frac{|\mathbf{l} \times \mathbf{l}'|}{|\mathbf{l}| |\mathbf{l}'|}. \quad (3.65)$$

Vektory  $\mathbf{l}$  a  $\mathbf{l}'$  jsou vektory, které představují element ve výchozí pozici a po natočení. Dopočte se ze vztahů

$$\mathbf{l} = (x_2 - x_1, 0) \quad (3.66)$$



Obrázek 3.9: Schéma natočení elementu

a

$$\mathbf{l}' = (x'_2 - x'_1, y'_2 - y'_1) = (x_2 + u_2 - (x_1 + u_1), y_2 + v_2 - (y_1 + v_1)). \quad (3.67)$$

Vektor  $\mathbf{l}$  představující výchozí vodorovnou pozici elementu není potřeba dopočítávat pro každý element a lze jej nahradit např. bázovým vektorem  $\mathbf{l} = (1, 0)$ . V případě, že výchozí geometrie není vodorovná, pak je nutné uvažovat rovněž bázový vektor  $\mathbf{l} = (1, 0)$ .

Významnou vlastností transformační matice je její ortogonalita. V důsledku to znamená, že její inverze se rovná její transpozici a vztah pro matici v globálním souřadném systému lze potom zapsat ve formě

$$\mathbf{K} = \mathbf{T}^\top \mathbf{K}_L \mathbf{T}, \quad (3.68)$$

$$\mathbf{K} = \mathbf{T}^\top (\mathbf{K}_T + \mathbf{K}_G) \mathbf{T} = \mathbf{T}^\top \mathbf{K}_T \mathbf{T} + \mathbf{T}^\top \mathbf{K}_G \mathbf{T}. \quad (3.69)$$

V posledním vztahu 3.69 se projeví dříve zmiňovaná výhoda při transformaci matice tuhosti. Matice geometrické tuhosti  $\mathbf{K}_G$  v uvedené podobě totiž zůstává při transformaci neměnná. To znamená, že je invariantní vůči pootočenému souřadnému systému a **není potřeba ji transformovat**. Vztah pro výslednou matici tuhosti v globálních souřadnicích lze poté zapsat ve formě

$$\mathbf{K} = \mathbf{T}^\top \mathbf{K}_T \mathbf{T} + \mathbf{K}_G. \quad (3.70)$$

Pro přehlednější zápis je dále zavedeno značení  $\sin(\varphi) = s$  a  $\cos(\varphi) = c$ . Po roznásobení lze získat vztah

$$\mathbf{K} = \frac{EA}{L_e} \begin{bmatrix} c^2 & cs & -c^2 & -cs \\ cs & s^2 & -sc & -s^2 \\ -c^2 & -sc & c^2 & sc \\ -cs & -s^2 & sc & s^2 \end{bmatrix} + \frac{EA(u_2 - u_1)}{L_e^2} \begin{bmatrix} 1 & 0 & -1 & 0 \\ 0 & 1 & 0 & -1 \\ -1 & 0 & 1 & 0 \\ 0 & -1 & 0 & 1 \end{bmatrix}. \quad (3.71)$$

Pro matici hmotnosti platí analogický vztah. Je-li však uvažována diagonální matice hmotnosti 3.50, pak je stejně jako matice geometrické tuhosti invariantní vůči pootočenému souřadnému systému a tudíž **není potřeba ji transformovat**. Hmotnost každého elementu je navíc po celou dobu výpočtu konstantní, z čehož plyne, že globální matici hmotnosti je potřeba sestavit pouze **jednou na začátku výpočtu a po celou dobu pak zůstává neměnná**.

Posledním členem pohybové rovnice popsaným v literatuře (např. v [2]), který je potřeba stanovit, je vektor vnitřních uzlových reakcí odpovídajících vzniklému napětí v prutu. Tento vektor má tvar

$$\mathbf{P} = N \begin{Bmatrix} -c \\ -s \\ c \\ s \end{Bmatrix}, \quad (3.72)$$

kde  $N$  je osová síla v prutu a určí se ze vztahu

$$N = EA\varepsilon = \frac{EA}{2} \left( \frac{L_e'^2 - L_e^2}{L_e^2} \right) \quad (3.73)$$

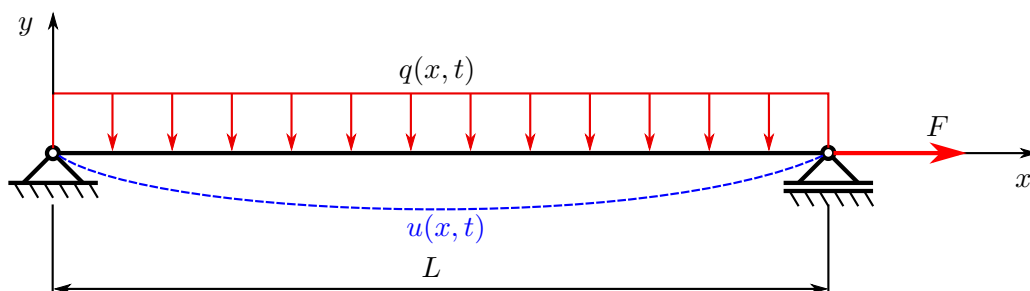
Výsledná pohybová rovnice v globálních souřadnicích má potom tvar

$$\mathbf{M} \ddot{\mathbf{u}}(t) + \mathbf{C} \dot{\mathbf{u}}(t) + \mathbf{K}(t) \mathbf{u}(t) = \mathbf{F}(t) - \mathbf{P}(t). \quad (3.74)$$

### 3.2.2a Příklad kmitání struny

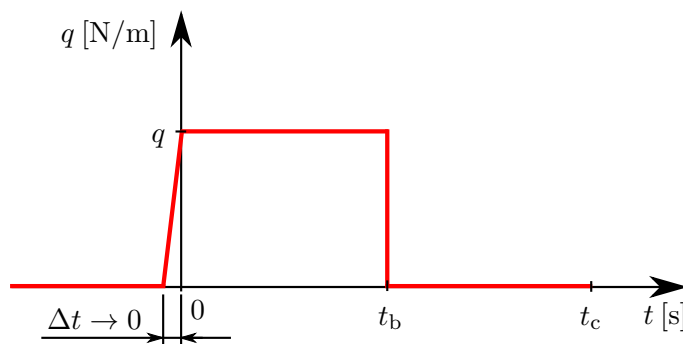
#### Popis úlohy

Ověření algoritmu rozšířeného do maticové formy je provedeno na příkladu kmitání předepjaté struny zatížené po délce konstantním spojitým zatížením  $q(x) = \text{konst.}$  Schéma úlohy je na obr. 3.10.



Obrázek 3.10: Schéma úlohy kmitání struny

Zadané parametry<sup>1</sup> struny jsou modul pružnosti  $E = 210 \text{ MPa}$ , hustota  $\rho = 7850 \text{ kg m}^{-3}$ , délka  $L = 1 \text{ m}$ , průměr  $d = 1 \text{ mm}$  a předepínací síla  $F = 4775 \text{ N}$ . Zatížení je aplikováno skokovou změnou v čase  $t = 0$  a v čase  $t_b = 0,05 \text{ s}$  je zatížení skokově odebráno, viz průběh zatížení na obr. 3.11. Velikost spojitého zatížení je  $q = 100 \text{ N/m}$ . Celkový čas analýzy je zvolen  $t_c = 0,1 \text{ s}$ .



Obrázek 3.11: Průběh zatížení struny

#### Analytické řešení

Výstupem analytického řešení kmitání je obecný vztah, který popisuje odezvu systému na buzení v kterémkoliv časovém okamžiku. Výsledný pohyb se skládá z partikulární a homogenní složky. Zde je analytické řešení omezeno pouze na stanovení velikosti statické výchylky struny, tedy partikulárního řešení. Diferenciální rovnice pro určení průhybu má tvar

$$-Fu''(x) = q(x). \quad (3.75)$$

<sup>1</sup>Parametry struny včetně předepínací síly byly zvoleny tak, aby její vlastní frekvence byla 440 Hz, což odpovídá např. frekvenci tónu *a*.

Řešením rovnice 3.75 pro okrajové podmínky  $u(0) = 0$ ,  $u(L) = 0$  a za předpokladu, že  $q(x) = \text{konst.}$  vyplyne analytický vztah průhybové čáry ve tvaru

$$u(x) = -\frac{1}{F} \left( q \frac{x^2}{2} - q \frac{L}{2} x \right). \quad (3.76)$$

Pro zadané hodnoty a dosazením za  $x = L/2$  se získá velikost statické výchylky středu struny  $u_{\text{stat}} = 2,6 \text{ mm}$ .

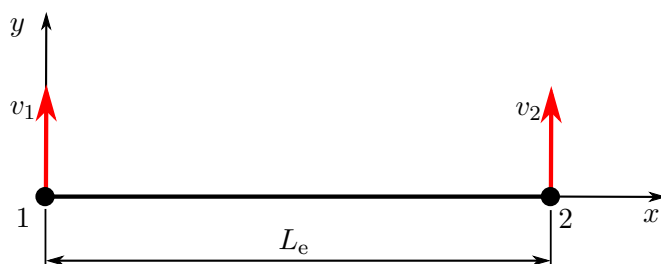
Vypočtená hodnota odpovídá partikulární složce výsledné výchylky struny a okolo této hodnoty bude struna po zatížení kmitat. Kmitavý pohyb popisuje homogenní složka, jejíž časový průběh lze získat např. pomocí Duhamelova integrálu. V [11] je odvozeno, že maximální výchylka při skokové změně zatížení je rovna dvojnásobku statické výchylky, tedy  $u_{\text{max}} = 2 u_{\text{stat}} = 5,2 \text{ mm}$ .

### Numerické řešení explicitní metodou

Cílem numerického řešení pomocí explicitního algoritmu je ověření maximální výchylky struny při skokové změně zatížení. Hlavními předpoklady pro numerické řešení předepjaté struny pomocí MKP jsou:

- 1 stupeň volnosti v každém uzlu
- malé deformace a posunutí
- předepínací síla je po celé délce struny konstantní
- konstantní průřez po celé délce struny.

Schéma použitého elementu pro diskretizaci struny se 2 stupni volnosti je na obr. 3.12.



Obrázek 3.12: Stupně volnosti elementu v lokálním souřadném systému

Přijetím těchto předpokladů se matice řádu  $4 \times 4$  odvozené v předchozí kapitole zredukuje na matice řádu  $2 \times 2$ , jejichž členy odpovídají 2. a 4. řádkům a jim příslušejícím sloupcům v původních maticích. Matice hmotnosti elementu má potom tvar

$$\mathbf{M} = \frac{\rho A L_e}{2} \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad (3.77)$$

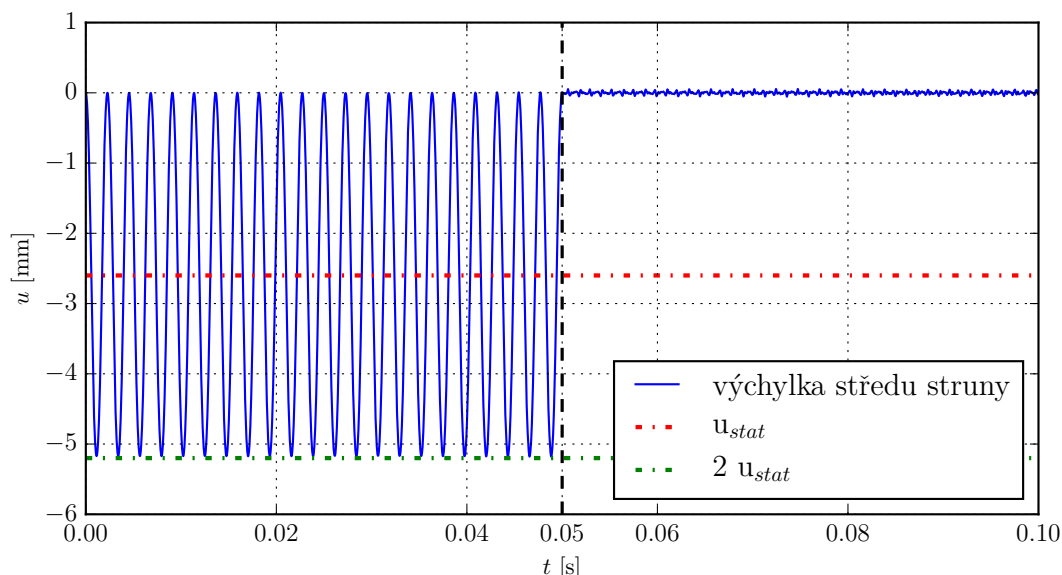
a výsledná matice tuhosti elementu, za předpokladu, že normálová síla ve vztahu 3.61 vzniklá od podélné deformace elementu je nahrazena předeplínací silou, tedy

$$\frac{EA(u_2 - u_1)}{L_e} = F = \text{konst.}, \text{ má tvar}$$

$$\mathbf{K} = \frac{F}{L_e} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix}. \quad (3.78)$$

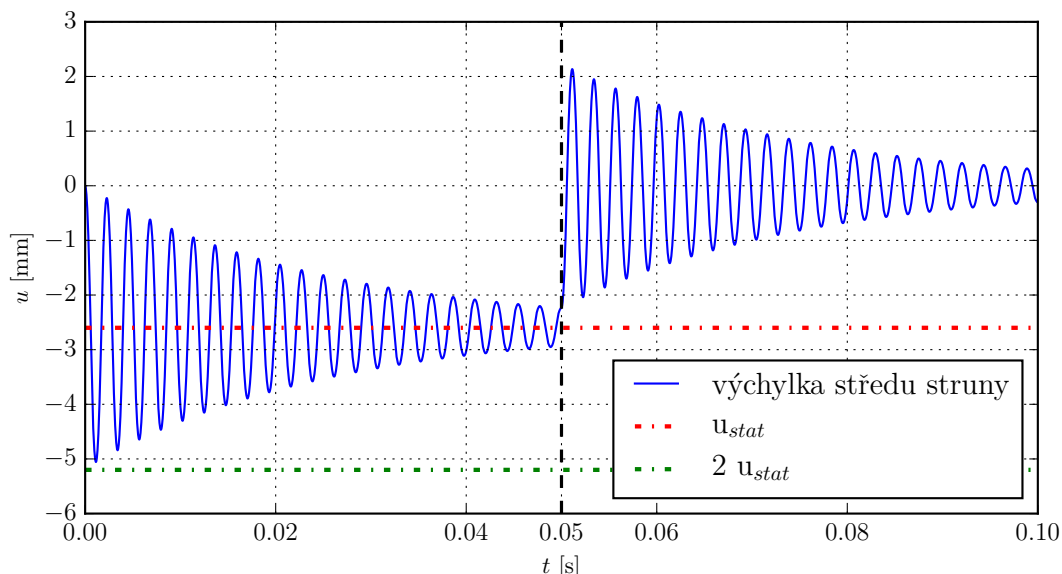
Vektor uzlových reakcí 3.72 bude při uvážení elementu z obr. 3.12 obsahovat pouze „sin“ složky. Za předpokladu malých deformací a posunutí, kdy  $\varphi \rightarrow 0$  však bude i  $\sin(\varphi) \rightarrow 0$  a vektor vnitřních uzlových reakcí bude tedy nulový.

Struna je diskretizována 80 elementy o délce  $L_e = 12,5$  mm. Výsledné průběhy jsou vykresleny pro střed struny. Z obr. 3.13 je patrné, že hodnota maximální výchylky se blíží hodnotě exaktní, která je znázorněna zeleně. Konkrétní vypočtená hodnota je  $u_{\max} = 5,18$  mm.



Obrázek 3.13: Časový průběh výchylky středového bodu struny

Pro názornost je zde uveden rovněž výpočet se zahrnutím vnějšího tlumení. Jak bylo uvedeno v kapitole 3.1, matice tlumení má tvar  $\mathbf{C} = \alpha \mathbf{M} + \beta \mathbf{K}$ , kde v tomto případě  $\beta = 0$  a  $\alpha = 80$ . Hodnota součinitele  $\alpha$  je záměrně zvolena tak, aby se tlumení na krátkém časovém úseku viditelně projevilo a bylo patrné ustalování výchylky na hodnotě statického průhybu, resp. na  $u = 0$  mm (viz obr. 3.14).



Obrázek 3.14: Časový průběh výchylky středového bodu struny - vliv tlumení

Porovnáním obr. 3.13 a 3.14 je na první pohled patrná rozdílná odezva po odlehčení struny. To lze vysvětlit následující úvahou. Celý výpočet lze v podstatě rozdělit na dvě samostatné části. První část odpovídá časovému intervalu  $\langle 0; t_b \rangle$  a druhá část intervalu  $\langle t_b; t_c \rangle$ . V prvním intervalu se řeší pohybová rovnice struny s nulovými počátečními podmínkami (nulová poč. výchylka a rychlost) a s nenulovým vektorem zatížení. Vypočtené hodnoty na konci prvního intervalu pak vstupují jako počáteční podmínky pro řešení pohybové rovnice ve druhém intervalu, kde je zátěžný vektor nulový.

V případě bez tlumení (obr. 3.13) dojde k odlehčení shodou okolností v okamžiku, kdy je výchylka i rychlost struny téměř nulová, tedy prakticky nulové počáteční podmínky pro druhý interval, a nedojde proto k výraznému rozkmitu. U tlumené struny (obr. 3.14) však dojde k odlehčení v momentě, kdy je rychlost stejná jako bez tlumení, tedy téměř nulová, avšak výchylka je výrazná. Následkem toho dojde k novému rozkmitání.

Vytvořený skript v Pythonu je k nalezení v příloze B.

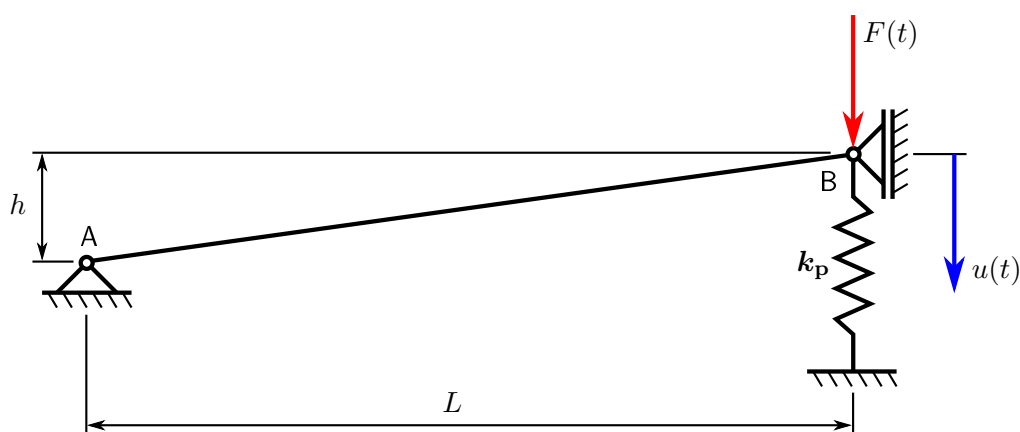
Na této úloze byla ověřena především funkčnost algoritmu v maticové podobě se zahrnutím vlivu tlumení.



## 3.2.2b Příklad kmitání systému prut-pružina

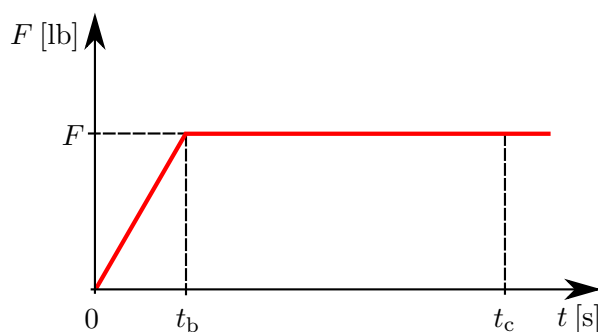
## Popis úlohy

Další příkladem je úloha dynamiky prutu uloženého na pružině, viz schéma na obr. 3.15. Cílem je vyšetření časového průběhu výchylky pohyblivého konce prutu B. Různé postupy řešení problému jsou uvedeny v [10] a [1]. Úloha je pro lepší srovnání s uvedenými zdroji řešena v angloamerických jednotkách.



Obrázek 3.15: Schéma úlohy kmitání systému prut-pružina

Horizontální vzdálenost mezi vazbami je  $L = 100$  in a vertikální vzdálenost  $h = 1$  in. Tuhostní parametr prutu je definován jako součin plochy průřezu a modulu pružnosti o velikosti  $EA = 10^7$  lb a hmotnostní parametr jako součin hustoty a plochy průřezu prutu o velikosti  $\rho A = 0,1$   $\text{lb s}^2/\text{in}^2$ . Tuhost pružiny má velikost  $k_p = 6$  lb/in. Zatížení je aplikováno s lineárním náběhem, přičemž maximální hodnoty  $F = 5$  lb je dosaženo v čase  $t_b = 0,1$  s; poté je velikost zatížení konstantní, viz průběh zatížení na obr. 3.16. Celkový čas analýzy je uvažován  $t_c = 8$  s.



Obrázek 3.16: Průběh zatížení prutu

## Analytické řešení

Analytické řešení je odvozeno v [10]. Řešení je analogické s řešením předchozí úlohy kmitání struny v kapitole 3.2.2a. Zatížení zde není aplikováno skokovou změnou, ale doba náběhu je vzhledem k daným parametrům soustavy tak malá, že **bude mít ve výsledku podobný účinek jako skoková změna**. Tato skutečnost se projeví na velikosti maximální výchylky bodu B, která bude dvojnásobná ve srovnání se statickou výchylkou. Řídící diferenciální rovnice, která popisuje odezvu systému na zatížení za daných počátečních podmínek má tvar

$$(F - ku) + \frac{AE}{2} \left[ \left( \frac{\cos(\theta_0)}{\cos(\theta)} \right)^2 - 1 \right] \cos(\theta) \tan(\theta) = \frac{\rho AL \ddot{u}}{3 \cos(\theta_0)}. \quad (3.79)$$

Úhel  $\theta_0$  je počáteční úhel natočení prutu od horizontální polohy a  $\theta$  je aktuální úhel natočení. Řešením diferenciální rovnice 3.79 se získá časový průběh výchylky bodu B. V [10] je tato rovnice řešena metodou Runge-Kutta a časový průběh posunutí je vykreslen do grafu. Zde jsou již uvedeny pouze číselné hodnoty. Hodnota maximální výchylky je  $u_{\max} = 1,52$  in, a je jí dosaženo v čase  $t = 3$  s.

## Numerické řešení explicitní metodou

Pro numerické řešení pomocí MKP byly přijaty následující předpoklady:

- diskretizace prutu pouze 1 elementem
- 2 stupně volnosti v každém uzlu
- velké natočení elementu.

Z uvedených předpokladů je zřejmé, že zde se již využijí matice odvozené v kapitolách 3.2.2 až 3.2.2 v nezměněné podobě. Jedinou výjimkou je matice hmotnosti elementu, která má v souladu s uvedenými zdroji tvar

$$\mathbf{M} = \frac{\rho AL_e}{3} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}. \quad (3.80)$$

Diskretizace prutu jediným elementem je zde nutná s ohledem na formulaci použitého prvku. Tyčový element není schopen přenést rotaci v uzlech a při spojení více elementů vznikne v uzlech kloub, ve kterém se mohou elementy volně natáčet. Možnou alternativou by bylo použití nosníkového prvku, který má v každém uzlu v rovině 3 stupně volnosti, a to dva posuvy ve dvou na sebe kolmých směrech a natočení uzlu. Pružina může být modelována jako dlouhý prut odpovídající tuhosti v tahu/tlaku, nebo může být její tuhost přímo aplikována do globální matice tuhosti. Obě varianty jsou rovnocenné a zde je využita druhá, jednodušší možnost. Předpokladem je, že při stlačení vzniká v pružině direkční síla pouze ve vertikálním

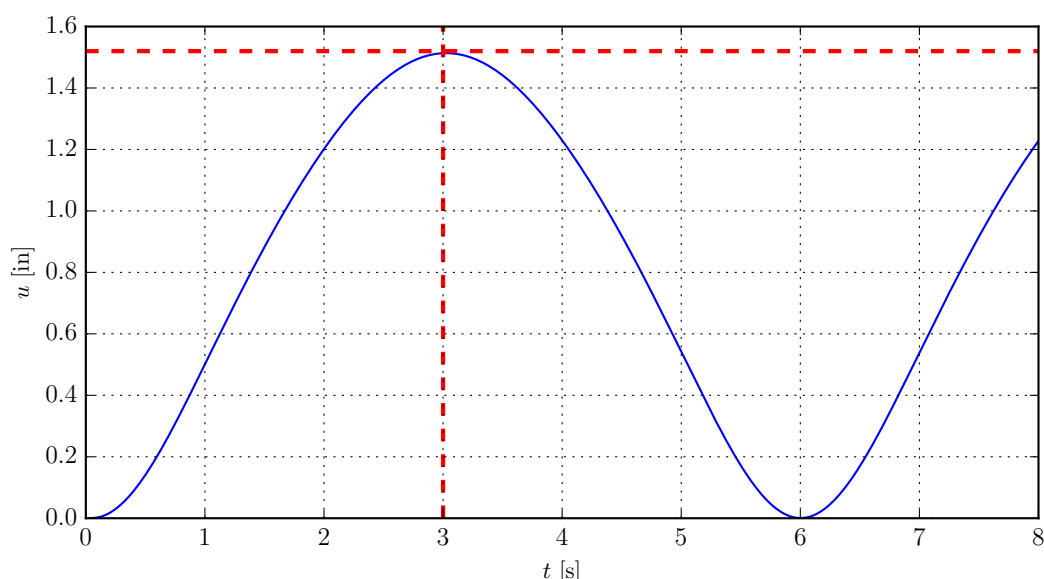
směru. Tento směr odpovídá 4. stupni volnosti elementu v globálních souřadnicích a tuhost pružiny  $k_p$  se tedy přičte do globální matice tuhosti elementu 3.71 na pozici  $[4, 4]$

$$\mathbf{K} = \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} \\ k_{21} & k_{22} & k_{23} & k_{24} \\ k_{31} & k_{32} & k_{33} & k_{34} \\ k_{41} & k_{42} & k_{43} & k_{44} + k_p \end{bmatrix}. \quad (3.81)$$

Vektor zatížení má po sestavení konečný tvar

$$\mathbf{F} = \begin{Bmatrix} 0 \\ 0 \\ 0 \\ -F \end{Bmatrix}. \quad (3.82)$$

Na obr. 3.17 je vykreslen časový průběh výchylky.



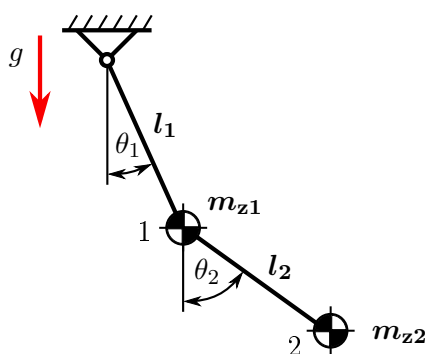
Obrázek 3.17: Časový průběh výchylky bodu B

Maximální dosažená hodnota výchylky je 1,51 in v čase 3 s, což je vynikající shoda s uvedenými zdroji. Výbornou shodu vykazuje rovněž průběh výchylky s [10]. Vytvořený skript je k nalezení v příloze C. Na této úloze bylo ověřeno chování definovaného elementu se čtyřmi stupni volnosti a správnost transformace lokální matice tuhosti do globálních souřadnic. V algoritmu bylo potřeba zavést použití polovičních hodnot posuvů při určování úhlu pro transformační matici a poloviční hodnoty tuhosti, obdobně jako uvádí autor v [10].

### 3.2.2c Kmitání dvojitého kyvadla

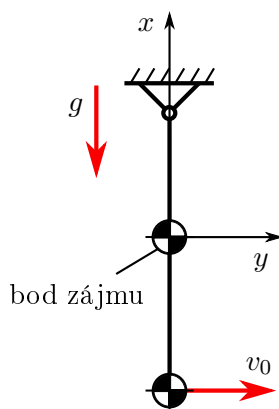
#### Popis úlohy

Pro další ověření správnosti dynamického chování a naladění algoritmu při větších výchylkách elementu byl zvolen příklad dvojitého kyvadla, pro který je známo analytické řešení. Tento systém vykazuje jev známý jako deterministický chaos. Pojem deterministický chaos znamená, že v definici chování systému nefiguruje žádná náhodná proměnná, ale je silně citlivý na změnu počátečních podmínek. Jedná se o úlohu tzv. matematického kyvadla, kde systém tvoří dva tuhé, nehmotné závěsy s hmotnými body  $m_{z1}$  a  $m_{z2}$  na volných koncích. Aplikováním počátečních podmínek, zde počáteční rychlosti koncového bodu 2, se kyvadlo rozkmitá. Zároveň na něj působí tíhové zrychlení. Schéma úlohy je na obr. 3.18.



Obrázek 3.18: Schéma úlohy dvojitého kyvadla

Uvažované parametry kyvadla jsou  $l_1 = l_2 = 0,15$  m a  $m_{z1} = m_{z2} = m_z = 1$  kg. Tíhové zrychlení je  $g = 9,81$  m s<sup>-2</sup>. Aplikace počáteční rychlosti je znázorněna na obr. 3.19 a její velikost je  $v_0 = 0,628$  m s<sup>-1</sup>. V této úloze bude dále analyzován pohyb bodu 1, proto je také souřadný systém zaveden do tohoto bodu<sup>2</sup>.



Obrázek 3.19: Aplikace počátečních podmínek dvojitého kyvadla

<sup>2</sup>Bod 1 je zvolen záměrně z důvodu zajímavějšího časového průběhu výchylek, rychlostí a zrychlení tohoto bodu. Průběhy v obou bodech však vykazují vynikající shodu analytického a numerického řešení.

### Analytické řešení

Analytické řešení je převzato z [9]. Chování systému je popsáno soustavou dvou diferenciálních rovnic ve tvaru

$$\ddot{\theta}_1 = -\frac{l_2}{\nu l_1} \ddot{\theta}_2 \cos(\theta_1 - \theta_2) - \frac{l_2}{\nu l_1} \dot{\theta}_2^2 \sin(\theta_1 - \theta_2) - \frac{g}{l_1} \sin(\theta_1) \quad (3.83)$$

$$\ddot{\theta}_2 = -\frac{l_1}{l_2} \ddot{\theta}_1 \cos(\theta_1 - \theta_2) + \frac{l_1}{l_2} \dot{\theta}_1^2 \sin(\theta_1 - \theta_2) - \frac{g}{l_2} \sin(\theta_2), \quad (3.84)$$

kde

$$\nu = 1 + (m_{z1} + m_{z2}).$$

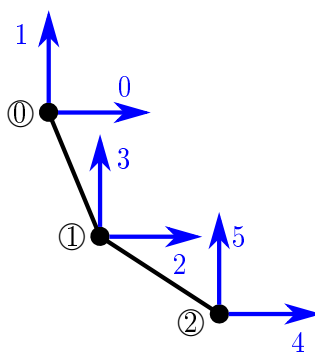
Proměnné  $\theta$ ,  $\dot{\theta}$  a  $\ddot{\theta}$  v rovnicích 3.83 a 3.84 jsou úhel natočení, úhlová rychlost a úhlové zrychlení závesů kyvadla,  $l$  jsou délky závesů a  $m_z$  hmotnosti na koncích závesů. Počáteční podmínka je zde úhlová rychlost  $\dot{\theta}_2$  druhého závesu, která se určí z počáteční rychlosti bodu 2 pomocí vztahu

$$\dot{\theta}_2 = \frac{v_0}{l_2}. \quad (3.85)$$

Soustava rovnic 3.83, 3.84 byla řešena s využitím Pythonu numericky převedením do stavového prostoru na soustavu diferenciálních rovnic 1. řádu. Výsledkem jsou trajektorie obou bodů a časové průběhy jejich výchylek, rychlostí a zrychlení. Vytvořený skript v Pythonu je k nalezení v příloze D.1.

### Numerické řešení explicitní metodou

Přestože se jedná o matematické kyvadlo, pro jehož analytické řešení stačí znát pouze délky závesů a hmotnosti hmotných bodů, pro numerické řešení je dále potřeba definovat i materiál závesů a jejich průřezové charakteristiky. Prakticky se tedy již řeší reálné kyvadlo na hmotném a pružném závesu, byť je jeho hmotnost oproti závažím zanedbatelná. Materiál závesu je ocel s modulem pružnosti  $E = 210$  GPa, Poissonovým číslem  $\mu = 0,3$  a plocha průřezu závesu je  $A = 1$  mm<sup>2</sup>.



Obrázek 3.20: Stupně volnosti diskretizovaného kyvadla

Obdobně jako u předchozího příkladu 3.2.2b je i zde každý záves diskretizován pouze jedním prvkem a rovněž jsou zde uvažovány matice odvozené v kapitolách 3.2.2 až 3.2.2. Na této úloze bude podrobněji rozebrán postup sestavení globálních matic soustavy. Použité

prvky mají v každém uzlu 2 stupně volnosti a jsou použity dva prvky (číslování uzlů, elementů a stupňů volnosti je od 0 pro zachování stejného zápisu, jako používá Python). Globální matice vzniknou sečtením prvků lokálních matic na odpovídajících pozicích, dle daného stupně volnosti, a znamená to, že globální matice systému budou mít rozměr 6x6. Globální matice hmotnosti tedy bude mít tvar

$$\mathbf{M} = \frac{1}{2} \begin{bmatrix} \textcolor{blue}{0} & \textcolor{blue}{1} & & \textcolor{blue}{2} & & \textcolor{blue}{3} & & \textcolor{blue}{4} & & \textcolor{blue}{5} \\ m_e^0 & 0 & & 0 & & 0 & & 0 & & 0 \\ 0 & m_e^0 & & 0 & & 0 & & 0 & & 0 \\ 0 & 0 & m_e^0 + m_e^1 + 2m_z & & 0 & & 0 & & 0 & \\ 0 & 0 & 0 & m_e^0 + m_e^1 + 2m_z & & 0 & & 0 & & 0 \\ 0 & 0 & 0 & 0 & m_e^1 + 2m_z & & 0 & & 0 & \\ 0 & 0 & 0 & 0 & 0 & m_e^1 + 2m_z & & 0 & & 0 \end{bmatrix} \begin{matrix} \textcolor{blue}{0} \\ \textcolor{blue}{1} \\ \textcolor{blue}{2} \\ \textcolor{blue}{3} \\ \textcolor{blue}{4} \\ \textcolor{blue}{5} \end{matrix}, \quad (3.86)$$

kde  $m_e$  jsou hmotnosti elementu (horní indexy značí číslo elementu). Jak již bylo zmíněno, tato matice zůstává po celou dobu výpočtu konstantní. Je patrné, že v každém uzlu je hmotnost v obou stupních volnosti stejná.

Globální matice tuhosti má po sestavení tvar

$$\mathbf{K} = \begin{bmatrix} \textcolor{blue}{0} & \textcolor{blue}{1} & & \textcolor{blue}{2} & & \textcolor{blue}{3} & & \textcolor{blue}{4} & & \textcolor{blue}{5} \\ k_{11}^0 & k_{12}^0 & & k_{13}^0 & & k_{14}^0 & & 0 & & 0 \\ k_{21}^0 & k_{22}^0 & & k_{23}^0 & & k_{24}^0 & & 0 & & 0 \\ k_{31}^0 & k_{32}^0 & k_{33}^0 + k_{11}^1 & k_{34}^0 + k_{12}^1 & k_{13}^1 & k_{14}^1 & & & & \\ k_{41}^0 & k_{42}^0 & k_{43}^0 + k_{21}^1 & k_{44}^0 + k_{22}^1 & k_{23}^1 & k_{24}^1 & & & & \\ 0 & 0 & k_{31}^1 & k_{32}^1 & k_{33}^1 & k_{34}^1 & & & & \\ 0 & 0 & k_{41}^1 & k_{42}^1 & k_{43}^1 & k_{44}^1 & & & & \end{bmatrix} \begin{matrix} \textcolor{blue}{0} \\ \textcolor{blue}{1} \\ \textcolor{blue}{2} \\ \textcolor{blue}{3} \\ \textcolor{blue}{4} \\ \textcolor{blue}{5} \end{matrix}. \quad (3.87)$$

Horní indexy zde opět značí číslo elementu a jednotlivé prvky  $k_{11}^0$  až  $k_{44}^1$  lokálních matic se určí v každém časovém kroku ze vztahu 3.71.

Vektor zatížení  $\mathbf{F}$  bude mít tvar

$$\mathbf{F} = \begin{pmatrix} 0 \\ -\frac{m_e^0}{2} g \\ 0 \\ -\left(\frac{m_e^0 + m_e^1}{2} + m_z\right) g \\ 0 \\ -\left(\frac{m_e^1}{2} + m_z\right) g \end{pmatrix} \begin{matrix} 0 \\ 1 \\ 2 \\ 3 \\ 4 \\ 5 \end{matrix} . \quad (3.88)$$

Velikost časového kroku se určí z rovnice 3.15. Maximální vlastní frekvence prvku  $\Omega_{\max}$  se určí modální analýzou nejmenšího prvku, resp. řešením problému vlastních čísel matice tuhosti a matice hmotnosti prvku. Vztah pro maximální vlastní frekvenci má potom tvar

$$\Omega_{\max} = 2\sqrt{\frac{A E}{m_{e \min} L_{e \min}}} \quad (3.89)$$

a výsledný vztah pro časový krok je ve tvaru

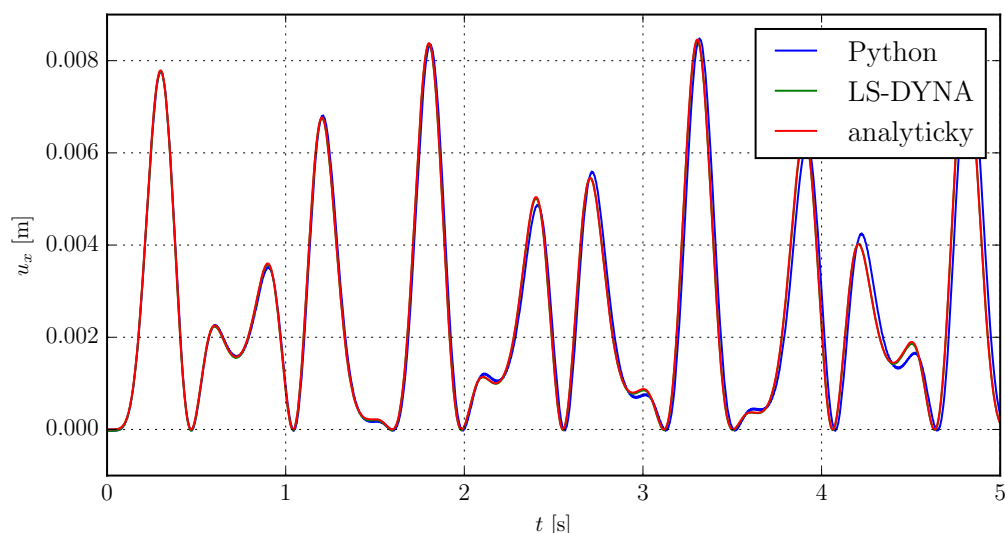
$$\Delta t = \eta \sqrt{\frac{m_{e \min} L_{e \min}}{A E}}, \quad (3.90)$$

kde  $m_{e \min}$  a  $L_{e \min}$  jsou hodnoty hmotnosti a délky příslušející nejmenšímu prvku sítě, za předpokladu, že plocha průřezu  $A$  a modul pružnosti  $E$  jsou konstantní.

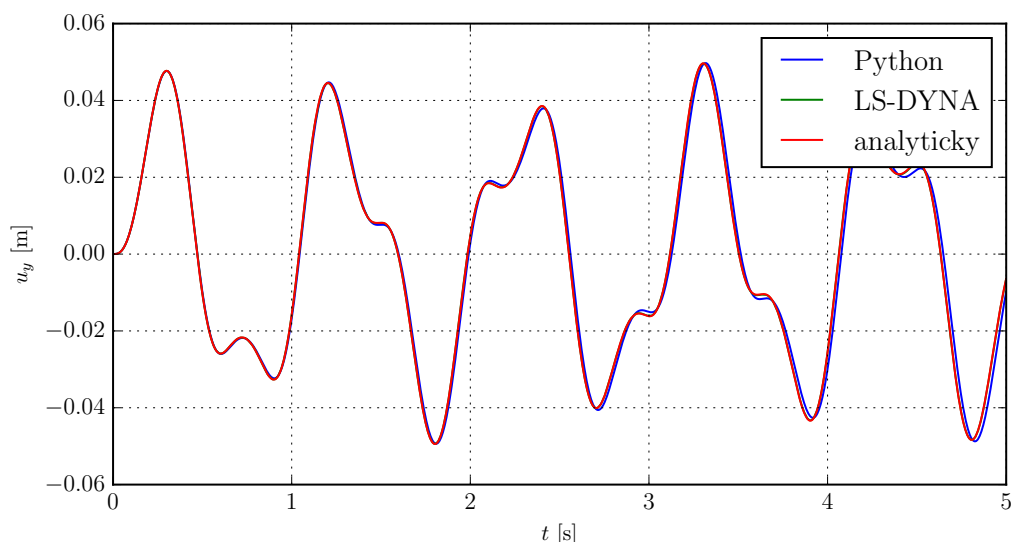
Pro výpočet časového kroku byl zvolen koeficient  $\eta = 0,9$ . Explicitní algoritmus pro řešení úlohy pomocí MKP v Pythonu je k nalezení v příloze D.2. Příklad byl rovněž řešen za pomoci komerčního softwaru LS-DYNA a následuje srovnání průběhů veličin.

### Průběh výchylek

Primárními neznámými jsou posuvy uzlů. Časové průběhy výchylek bodu 1 v ose x a y (viz obr. 3.19) jsou vykresleny na obr. 3.21 a 3.22. V grafech jsou srovnány průběhy posuvu bodu z numerického řešení analytických pohybových rovnic 3.83 a 3.84, MKP řešení pomocí explicitního algoritmu v Pythonu a řešení z LS-DYNA.



Obrázek 3.21: Časový průběh výchylky bodu 1 v ose x



Obrázek 3.22: Časový průběh výchylky bodu 1 v ose y

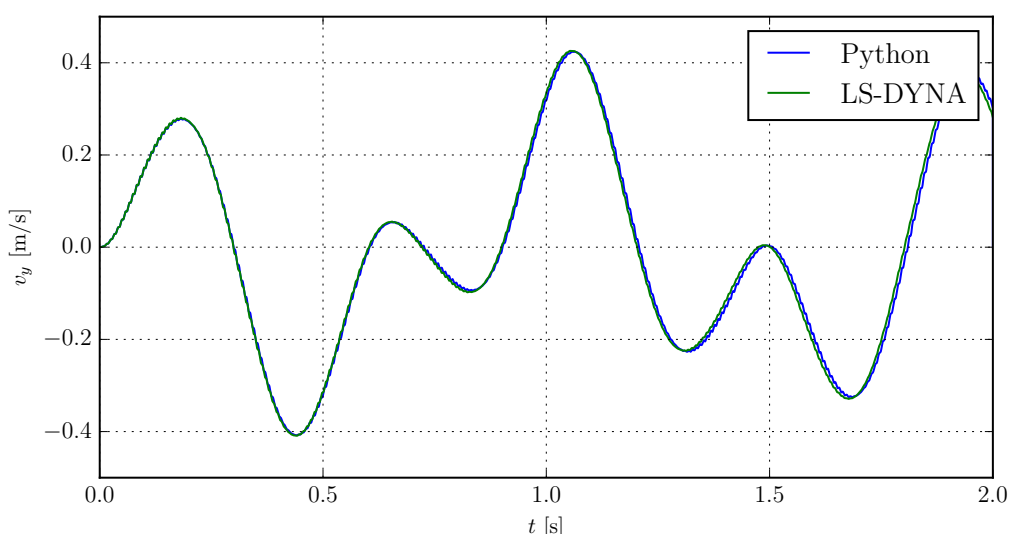
Z obr. 3.21 a 3.22 je patrné, že řešení analytických rovnic v podstatě splývá s řešením



úlohy v LS-DYNA. Průběhy získané explicitním výpočtem v Pythonu vykazují výbornou shodu. Od času přibližně 2 s je již pozorovatelná mírná odchylka, ale jelikož se délka explicitních analýz pohybuje převážně v řádu stovek milisekund, lze získaný průběh považovat za více než dostačující.

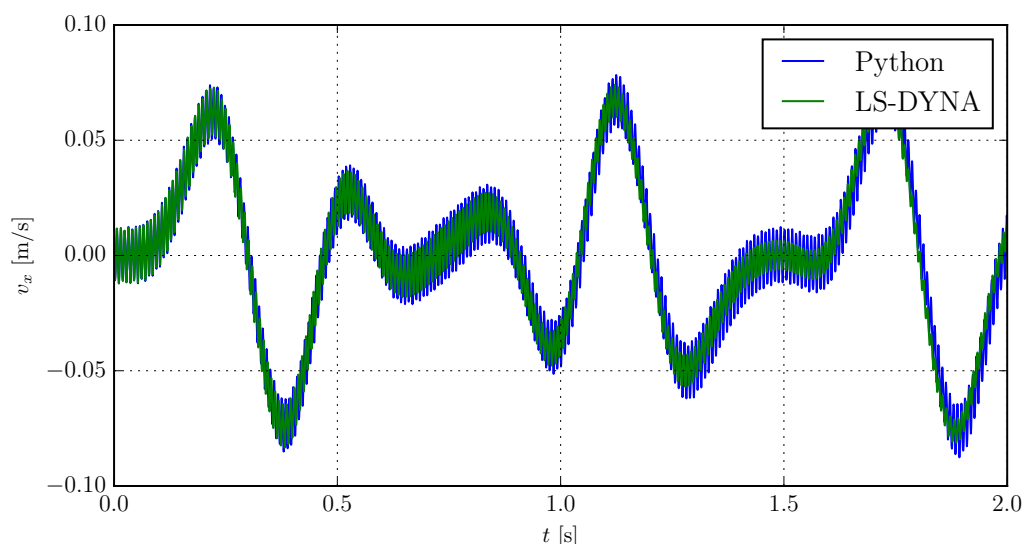
### Průběh rychlostí

Rychlost pohybu uzlů je již potřeba dopočíst ze známých posuvů pomocí vztahu 3.4. Následuje srovnání rychlostí bodu 1 v ose x a y z LS-DYNA a explicitního výpočtu v Pythonu. Analytické řešení již není uvedeno, jelikož se shoduje s řešením z LS-DYNA.

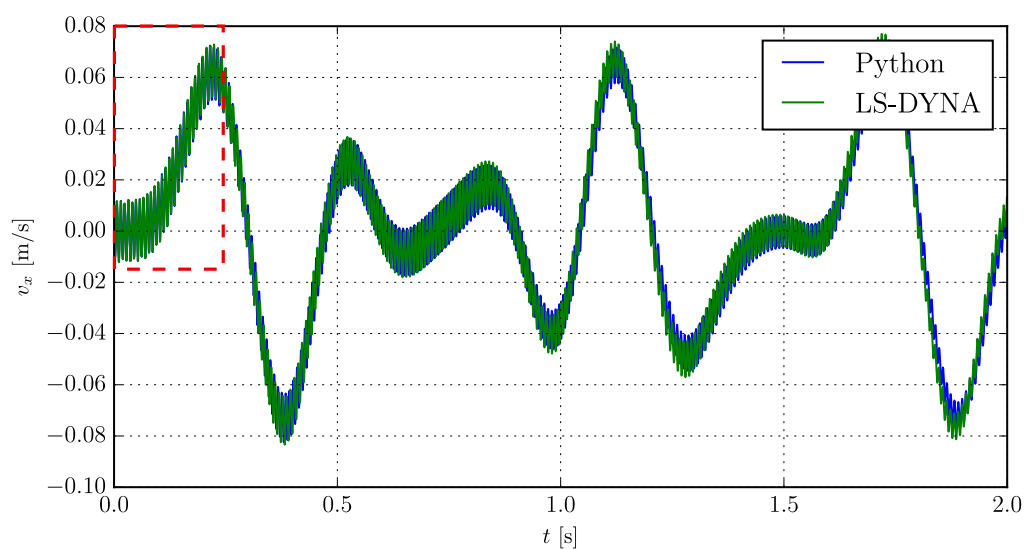


Obrázek 3.23: Časový průběh rychlosti bodu 1 v ose y

Je patrné, že rychlosti v ose y (obr. 3.23) vykazují opět výbornou shodu. Na průběhu rychlosti v ose x (obr. 3.24) se navíc projevuje pružení závěsu. Je zřejmé, že algoritmus použitý v softwaru LS-DYNA je určitým způsobem modifikován a tato modifikace se projevuje jako vnitřní (materiálové) tlumení, díky kterému vibrace po čase ustanou. Použití systémového tlumení a dynamické relaxace uvádí rovněž teoretický manuál LS-DYNA [14]. V Pythonu dosud nebyla uvažována žádná forma tlumení. Bylo proto provedeno několik analýz s uvažováním materiálového tlumení ve formě matice  $\mathbf{C} = \beta \mathbf{K}$  s cílem určit velikost konstanty  $\beta$ . Hodnota součinitele byla stanovena přibližně na  $\beta = 3 \cdot 10^{-6}$ . Na obr. 3.25 je pak vykreslen průběh rychlosti v ose x se zahrnutým vnitřním tlumením.

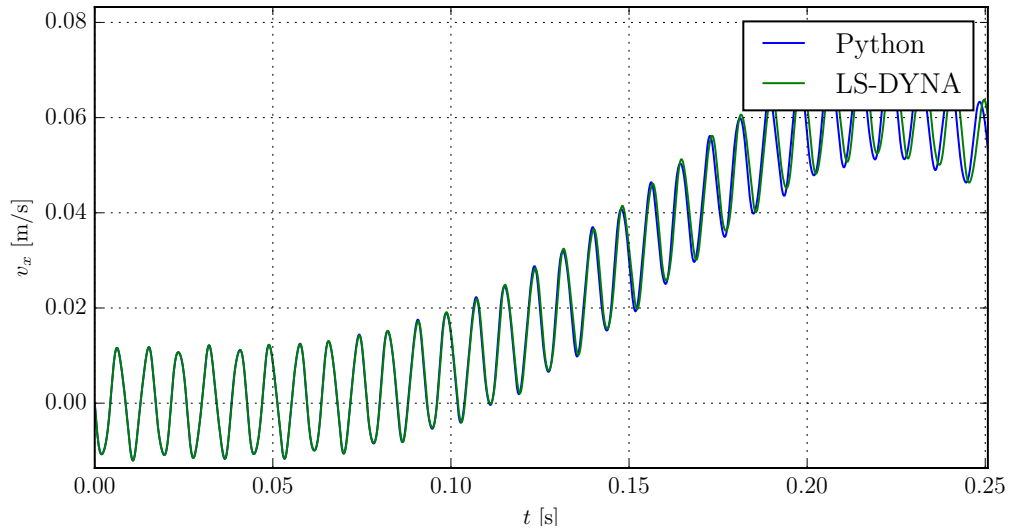


Obrázek 3.24: Časový průběh rychlosti bodu 1 v ose x



Obrázek 3.25: Časový průběh rychlosti bodu 1 v ose x s tlumením

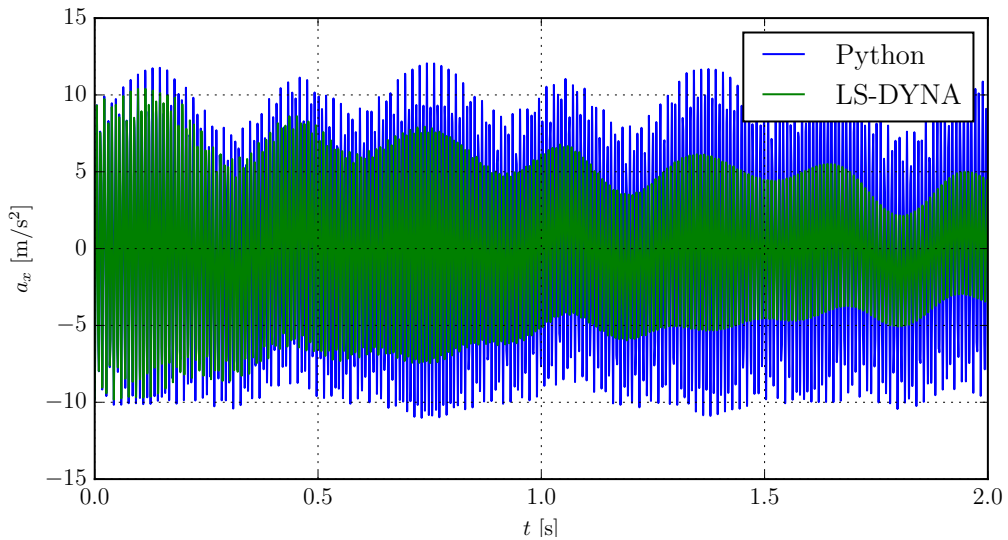
Detail tlumeného průběhu rychlosti bodu 1 v ose x je na obr. 3.26.



Obrázek 3.26: Časový průběh rychlosti bodu 1 v ose x s tlumením - detail

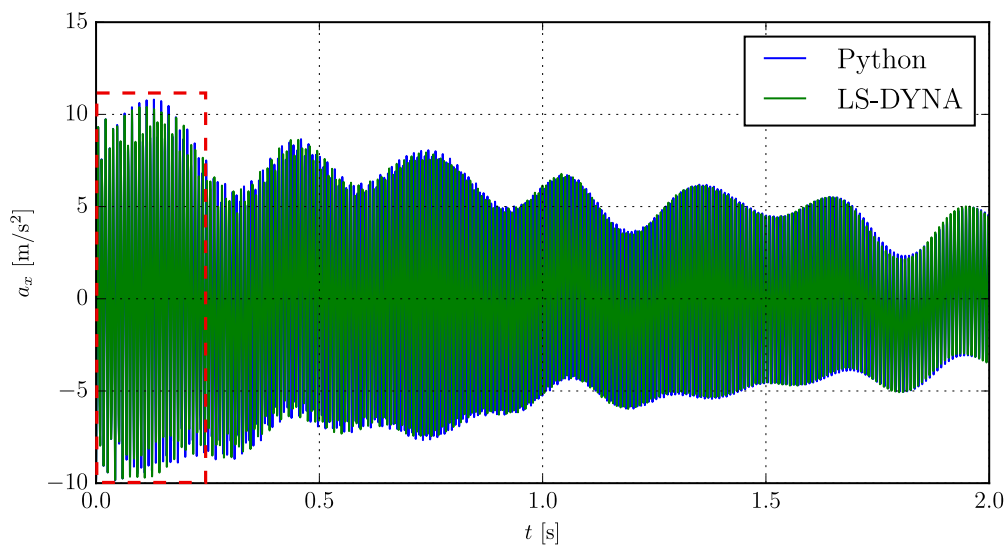
### Průběh zrychlení

Rovněž zrychlení uzlů je potřeba dopočítat z posuvů a sice pomocí vztahu 3.5. Na obr. 3.27 a 3.30 jsou vykresleny netlumené průběhy zrychlení bodu 1 v ose x a y. Pro tyto průběhy platí obdobné zjištění jako pro průběhy rychlosti v těchto směrech. Je zde však více patrný rozdíl mezi algoritmem v Pythonu bez tlumení a algoritmem LS-DYNA.

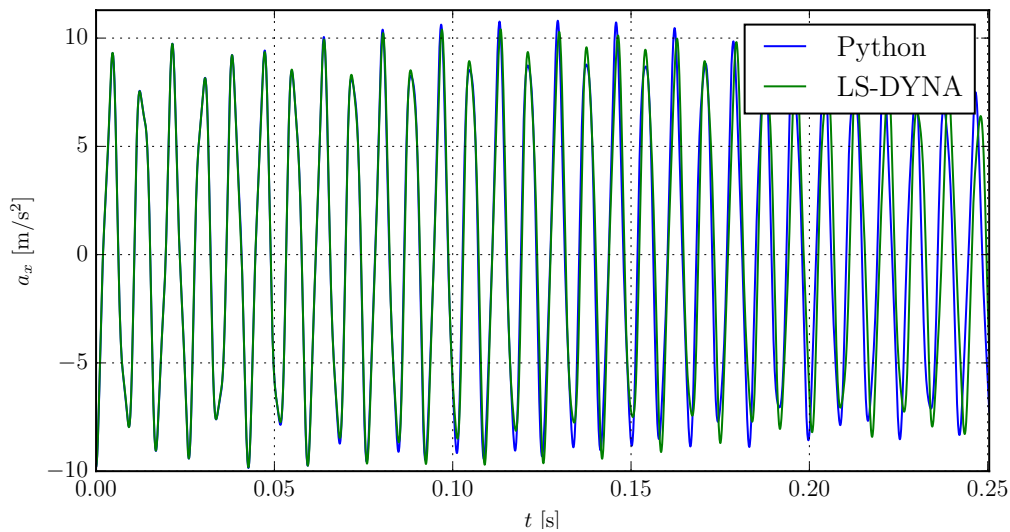


Obrázek 3.27: Časový průběh zrychlení bodu 1 v ose x

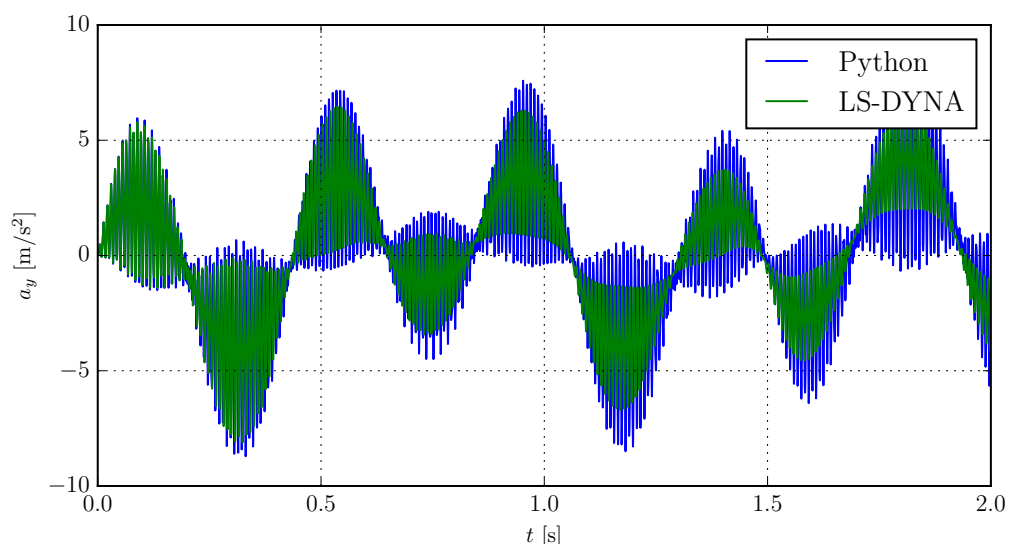
Na obr. 3.28 je již vykreslen tlumený průběh zrychlení v ose x a v detailu pak na obr. 3.29.



Obrázek 3.28: Časový průběh zrychlení bodu 1 v ose x s tlumením

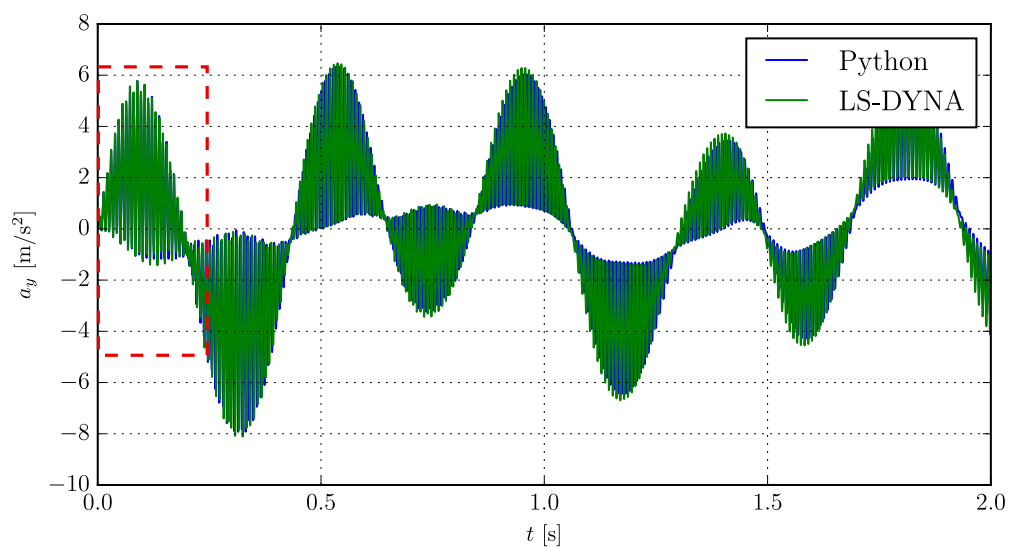


Obrázek 3.29: Časový průběh zrychlení bodu 1 v ose x s tlumením - detail

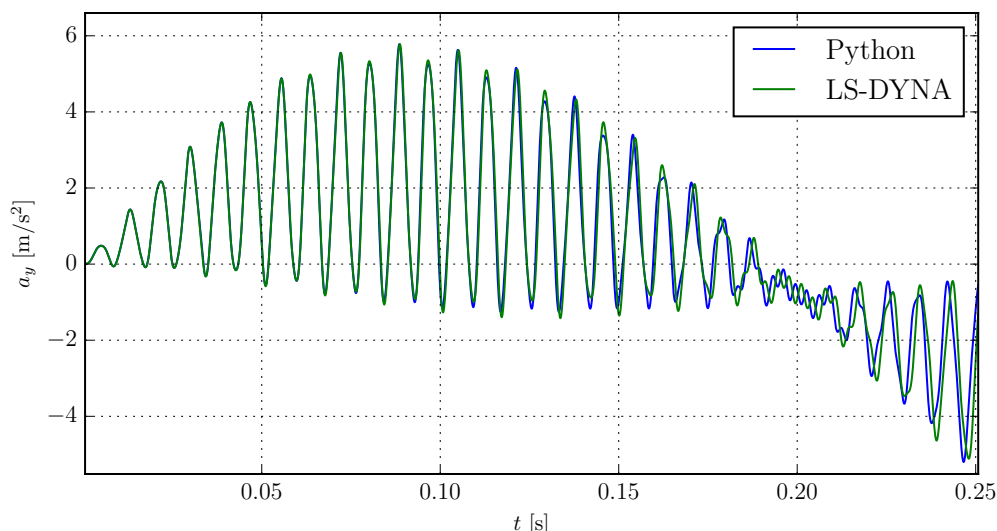


Obrázek 3.30: Časový průběh zrychlení bodu 1 v ose y

Časový průběh zrychlení v ose y s tlumením je vykreslen na obr. 3.31 a v detailu na obr. 3.32.



Obrázek 3.31: Časový průběh zrychlení bodu 1 v ose y s tlumením



Obrázek 3.32: Časový průběh zrychlení bodu 1 v ose y s tlumením - detail

Ze všech uvedených průběhů a z detailů na obr. 3.26, 3.29 a 3.32 je patrná vynikající shoda výsledků získaných výpočtem pomocí vlastního algoritmu s výsledky z komerčního softwaru LS-DYNA. Tímto příkladem byly verifikovány veškeré postupy a vztahy odvozené v předchozích kapitolách. Navíc zde bylo srovnáním s LS-DYNA stanovena přibližná hodnota vnitřního tlumení, které má vliv na stabilitu pozdějších výpočtů. S takto naladěným algoritmem bude dále provedena vlastní zjednodušená analýza bezpečnostního pásu.

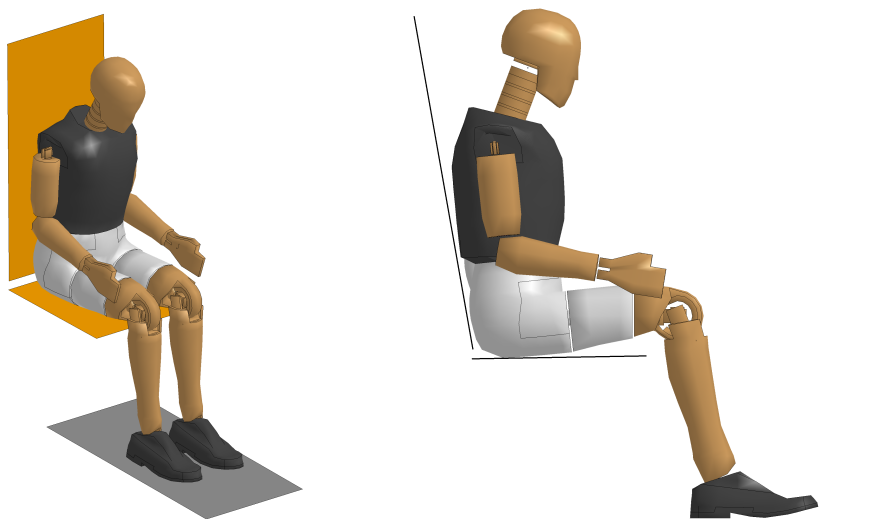
## Kapitola 4

# Simulace dynamického chování bezpečnostních pásů

V této kapitole jsou provedeny numerické simulace nárazové zkoušky leteckého sedadla dle [19] pomocí dvou přístupů. V první části je k analýze využit vlastní verifikovaný kód v jazyce Python, ve druhé části pak komerční software LS-DYNA. Oba přístupy využívají stejný výchozí prostorový model úlohy, který je popsán v následujícím textu. Pro analýzu vlastním algoritmem je však model využit pouze k získání geometrie pásu.

### Výchozí model úlohy

Pomocí volně dostupného softwaru pro přípravu modelu a zpracování výsledků LS-PrePost společnosti LSTC byl vytvořen 3D model úlohy, který je tvořen sedadlem, podlahou a testovací figurínou, viz obr. 4.1. Podrobnější popis modelu, včetně použitých prvků,

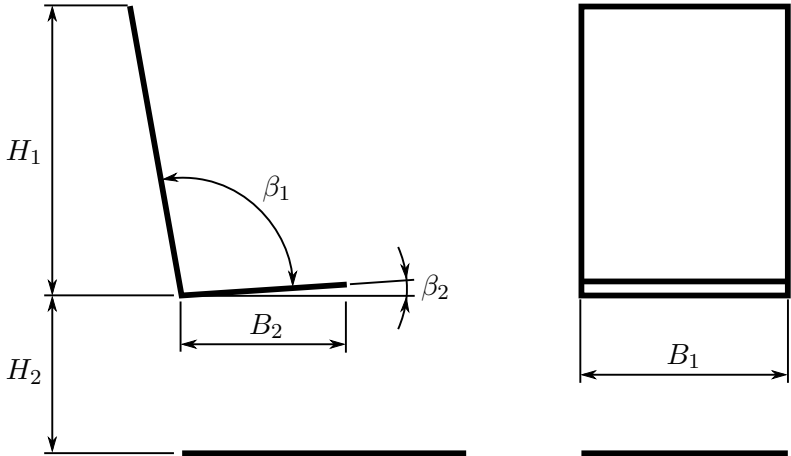


Obrázek 4.1: 3D výchozí model

materiálů a definic kontaktů je uveden dále v kapitole 4.2.

### Rozměry modelu sedadla a podlahy

Sedadlo je v ranné fázi vývoje a jeho podrobná geometrie v současné době není známa. Pro určení reakcí v kotevních bodech pásu byl proto vytvořen zjednodušený model, který je tvořen dvěma absolutně tuhými deskami představujícími sedák a opěrnou část sedadla. Rozměry tohoto modelu jsou patrné z obr. 4.2 a jejich hodnoty jsou uvedeny v tab. 4.1.



Obrázek 4.2: Rozměry modelu sedadla

$H_1$ [mm]	$H_2$ [mm]	$B_1$ [mm]	$B_2$ [mm]	$\beta_1$ [°]	$\beta_2$ [°]
645	408	450	440	99	1

Tabulka 4.1: Rozměry modelu sedadla

Model podlahy tvoří rovněž absolutně tuhá deska. Jak již bylo uvedeno v textu dříve, podlaha nemusí být v testu ani v simulaci zahrnuta a je zde především pro lepší vizualizaci testu a přehlednost.

### Model figuríny

Dle požadavků [19] je použita 50 percentilová mužská testovací figurína, jejíž kalibrovaný model byl poskytnut společností LSTC, na základě platné licence. Přesné označení použité verze je LSTC Hybrid III 50<sup>th</sup> Percentile Fast Dummy Model verze 2.0 (viz [17]). Obsahem je polohovatelná figurína včetně definovaných materiálů a základních kontaktů. Tento model je určen k simulaci chování lidského těla při nárazech a je možné díky němu vyhodnotit např. kritéria poranění, které jsou primárními výsledky převážné většiny simulací crash testů.



## 4.1 Analýza bezpečnostního pásu pomocí vlastního algoritmu

Analýza spočívá ve vytvoření diskretizovaného modelu pásu, aplikaci předepsaných okrajových podmínek dle [19] a stanovení časového průběhu reakcí v kotevních bodech pásu za pomoci vlastního explicitního algoritmu implementovaného v jazyce Python.

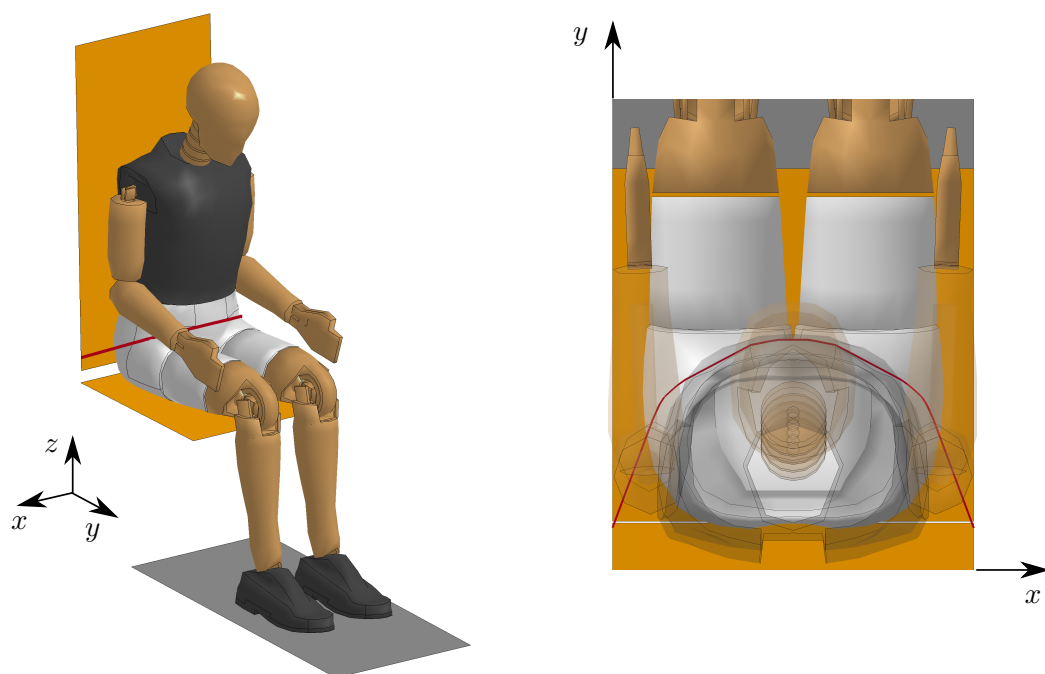
### Předpoklady

Při řešení byly přijaty následující předpoklady a zjednodušení:

- úloha je řešena jako rovinná
- hmotnost figuríny je rozložena lineárně po délce pásu a je aplikována do jednotlivých uzlů
- materiál je uvažován jako lineární (platí Hookeův zákon)
- vliv gravitace je zanedbán

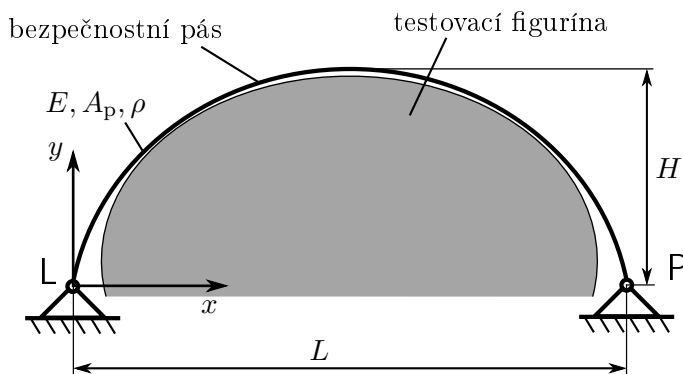
#### 4.1.1 Model pásu

Zjednodušená geometrie pásu, v podobě křivky obepínající pánev sedící figuríny, je vytvořena na výchozím modelu, popsaném v kapitole 4. Křivka představuje střednici skutečného pásu. Vizualizace modelu se zjednodušeným pásem je na obr. 4.3. Geometrie pásu je označena červeně.



Obrázek 4.3: 3D výpočetní model se zjednodušeným modelem pásu; vpravo pohled shora

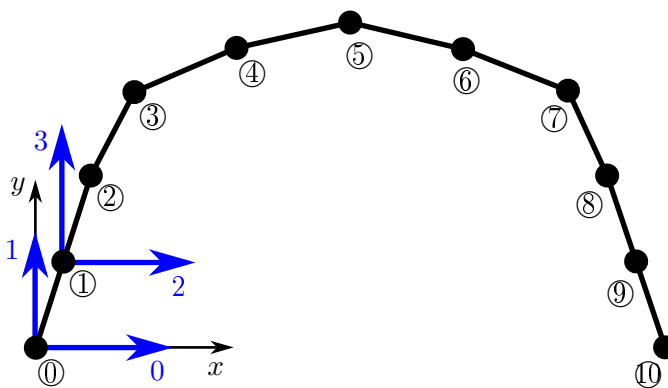
Tato křivka je diskretizována 10 elementy o délce  $L_e \approx 80$  mm. Geometrie pásu byla sklopena do vodorovné roviny x-y a nové souřadnice uzlů tvoří výchozí geometrii pro vlastní analýzu. Schéma úlohy je na obr. 4.4 a odpovídá uspořádání na obr. 4.3 vpravo, při pohledu shora.



Obrázek 4.4: Geometrie pásu pro zjednodušený výpočet

Uvažované materiálové parametry nylonového pásu jsou uvedeny v tab. 2.3. Hlavními rozměry modelu jsou rozteč kotevních bodů  $L = 450$  mm a výška  $H = 260$  mm. Průřez profilu pásu a jeho rozměry jsou popsány v kapitole 2.5. Kotevní body jsou označeny pro lepší orientaci L a P.

Výstupem výše uvedeného postupu tvorby geometrie je textový soubor se souřadnicemi uzlů pásu exportovaný z 3D modelu. Po načtení sítě algoritmem lze vykreslit skutečnou geometrii, viz obr. 4.5. Na obrázku je již znázorněn systém značení uzlů sítě a stupňů volnosti v globálním souřadném systému (značeny modře). Stupně volnosti, uzly i elementy jsou číslovány od nuly, pro zachování konzistentního zápisu s kódem v jazyce Python, který čísluje prvky matic, vektorů, apod. rovněž od nuly.



Obrázek 4.5: Skutečná diskretizovaná geometrie pásu

### 4.1.2 Rozložení hmotnosti figuríny do uzlů pásu

Jak již bylo zmíněno v předpokladech zjednodušené analýzy, hmotnost figuríny je lineárně rozložena do uzlů diskretizovaného pásu. Lineární rozložení po celé délce pásu se ukázalo jako nejvhodnější pro stabilitu výpočtu. Rozložení hmotnosti na intervalu  $\langle 0; L/2 \rangle$  je popsáno lineární funkcí

$$m_i = k i \quad (4.1)$$

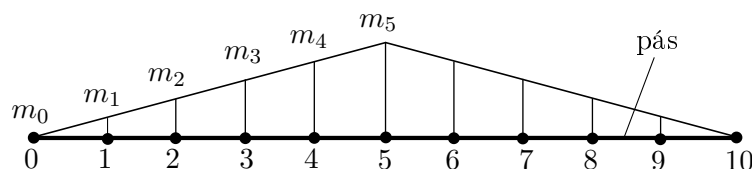
a na intervalu  $\langle L/2; L \rangle$  funkcí

$$m_i = q - k i, \quad (4.2)$$

kde  $m_i$  je dílčí hmotnost přidaná do uzlu,  $k$  a  $q$  jsou hmotnostní parametry o velikosti  $k = 3,08 \text{ kg}$  a  $q = 30,8 \text{ kg}$  a  $i$  je číslo uzlu  $i = 0, 1, 2, \dots, 10$ . Součet dílčích hmotností je roven celkové hmotnosti figuríny

$$m_f = \sum_{i=0}^{10} m_i = 77 \text{ kg}. \quad (4.3)$$

Na obr. 4.6 je znázorněno rozložení hmotnosti testovací figuríny do uzlů diskretizovaného bezpečnostního pásu. V tab. 4.2 jsou uvedeny konkrétní číselné hodnoty.



Obrázek 4.6: Schéma rozdělení hmotnosti figuríny do uzlů na rozvinutém pásu

$i$ [-]	0	1	2	3	4	5	6	7	8	9	10
$m_i$ [kg]	0	3,08	6,16	9,24	12,32	15,4	12,32	9,24	6,16	3,08	0

Tabulka 4.2: Rozložení hmotnosti figuríny

Uvedené dílčí hmotnosti jsou uzlové příspěvky do globální matice hmotnosti soustavy a aplikují se přičtením na odpovídající pozici, dle daného stupně volnosti na hlavní diagonálu matice, jak je znázorněno ve vztahu 4.4

$$\mathbf{M} = \begin{bmatrix} \tilde{m}_{00} + m_0 & 0 & 0 & 0 & \cdots & 0 \\ & \tilde{m}_{11} + m_0 & 0 & 0 & \cdots & 0 \\ & & \tilde{m}_{22} + m_1 & 0 & \cdots & 0 \\ & & & \tilde{m}_{33} + m_1 & \cdots & 0 \\ & & & & \ddots & \vdots \\ & & & & & \tilde{m}_{21\ 21} + m_{10} \end{bmatrix}, \quad (4.4)$$

kde  $\tilde{m}_{00} = \tilde{m}_{11} = \frac{m_e^0}{2}$ ,  $\tilde{m}_{22} = \tilde{m}_{33} = \frac{m_e^0 + m_e^1}{2}$ ,  $\tilde{m}_{44} = \tilde{m}_{55} = \frac{m_e^1 + m_e^2}{2}$ , atd. Horní index u hmotnosti prvku značí číslo elementu, takže např.  $m_e^0$  je hmotnost nultého elementu, atd. Je zřejmé, že v každém uzlu je celková hmotnost pro oba stupně volnosti stejná. Pro přehlednost jsou proto výsledné prvky matice hmotnosti dále značeny pomocí indexů uzlů, kterým dané hmotnosti náleží

$$\mathbf{M} = \begin{bmatrix} \overline{m}_0 & 0 & 0 & 0 & \cdots & 0 \\ & \overline{m}_0 & 0 & 0 & \cdots & 0 \\ & & \overline{m}_1 & 0 & \cdots & 0 \\ & & & \overline{m}_1 & \cdots & 0 \\ \text{Sym.} & & & & \ddots & \vdots \\ & & & & & \overline{m}_{10} \end{bmatrix}. \quad (4.5)$$

#### 4.1.3 Okrajové a počáteční podmínky

Počáteční podmínky, tedy počáteční výchylka, počáteční rychlost a počáteční zatížení, jsou u této úlohy nulové. Dále jsou popsány aplikované okrajové podmínky.

Uchycení pásu je realizováno pomocí pevných kloubů na obou koncích. Na celý pás, tj. do každého uzlu, je aplikováno časově proměnné zatížení odpovídající, dle 2. Newtonova pohybového zákona, součinu hmotnosti v uzlu a jeho zrychlení

$$\mathbf{F}_i(t) = \overline{m}_i \mathbf{a}(t). \quad (4.6)$$

Rozepsáním rovnice 4.6 do složek, dle zavedeného souřadného systému, se získají prvky vektoru zatížení ve tvaru

$$F_{ix}(t) = -\overline{m}_i a(t) \sin(\alpha_2) \quad (4.7)$$

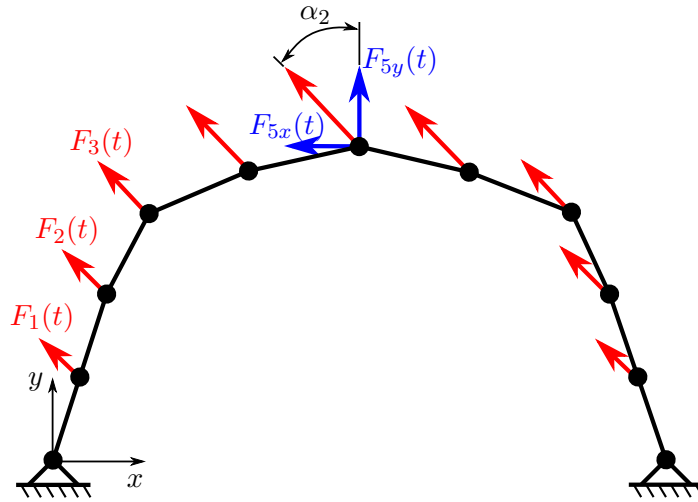
$$F_{iy}(t) = \overline{m}_i a(t) \cos(\alpha_2) \quad (4.8)$$

pro číslo uzlu  $i = 0, 1, 2, \dots, 10$ . Velikost  $a(t)$  odpovídá předepsanému zpomalení (obr. 1.1). Úhel  $\alpha$  je minimální úhel natočení sedadla od podélné osy letadla, požadovaný normou [19], o velikosti  $\alpha_2 = 10^\circ$ . Schématicky je aplikace okrajových podmínek znázorněna na obr. 4.7.

Vektor vnějších sil  $\mathbf{F}(t)$  v rovnici 3.74 má po sestavení tvar

$$\mathbf{F}(t) = \begin{Bmatrix} -\overline{m}_0 a(t) \sin(\alpha_2) \\ \overline{m}_0 a(t) \cos(\alpha_2) \\ -\overline{m}_1 a(t) \sin(\alpha_2) \\ \overline{m}_1 a(t) \cos(\alpha_2) \\ \vdots \\ \overline{m}_{10} a(t) \cos(\alpha_2) \end{Bmatrix} \quad (4.9)$$

Hodnota časového kroku se určí opět ze vztahu 3.90 a např. pro uvažované  $\eta = 1$  má hodnotu  $\Delta t \approx 7,967 \cdot 10^{-5}$  s.



Obrázek 4.7: Zatížení pásu

#### 4.1.4 Výsledky analýzy

Výsledkem analýzy pomocí vlastního algoritmu jsou průběhy reakcí v kotevních bodech pásu. Jejich hodnota se dopočítává podle vztahu

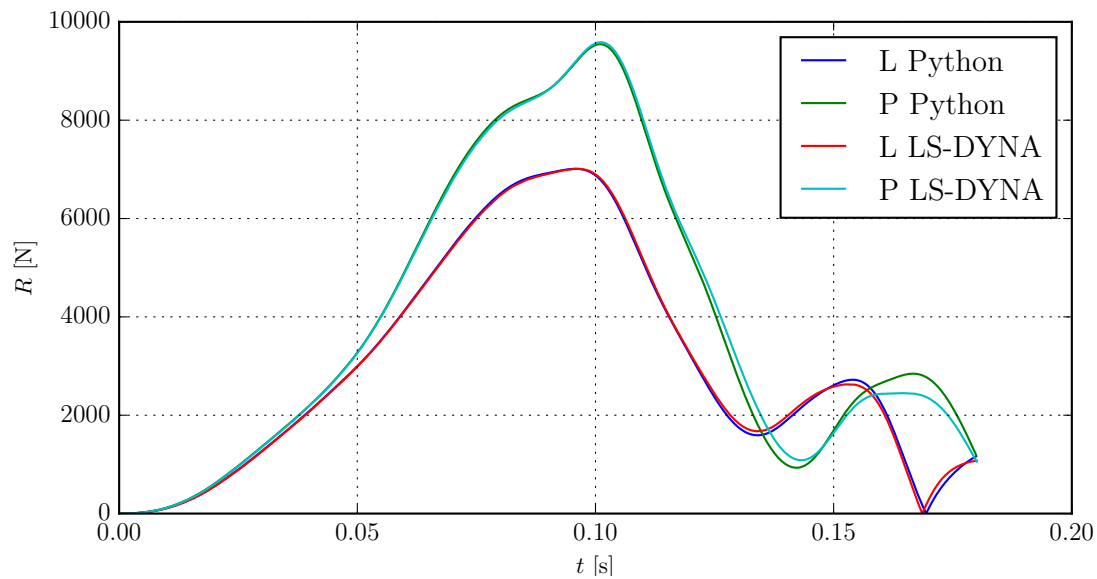
$$\mathbf{R}(t) = \mathbf{K}(t)\mathbf{u}(t) + \mathbf{P}(t). \quad (4.10)$$

Výsledné reakce se skládají ze dvou složek ve směru souřadných os  $R_x$  a  $R_y$  podle vztahu

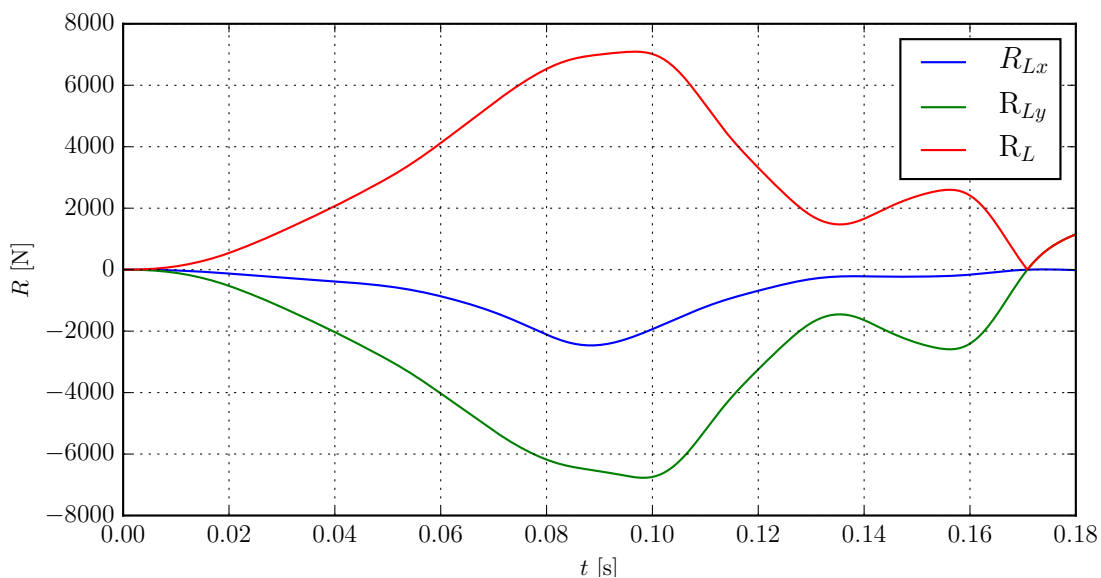
$$R(t) = \sqrt{R_x(t)^2 + R_y(t)^2}. \quad (4.11)$$

Časový průběh výslednic reakcí je vykreslen na obr. 4.8. Stejná úloha byla pro srovnání řešena i komerčním softwarem LS-DYNA a v grafu je rovněž vykreslen průběh reakcí získaný touto analýzou.

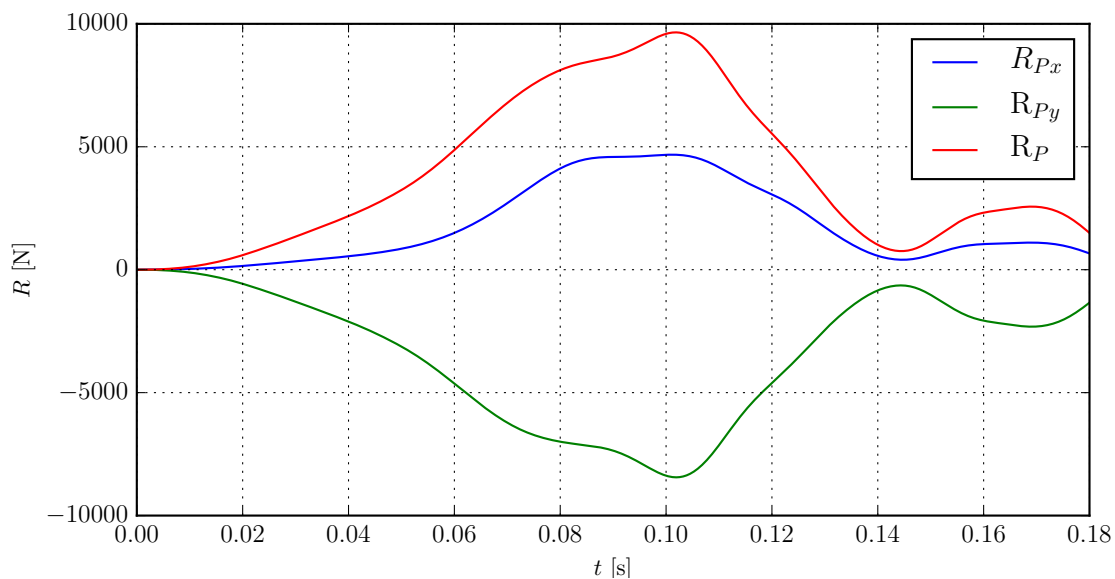
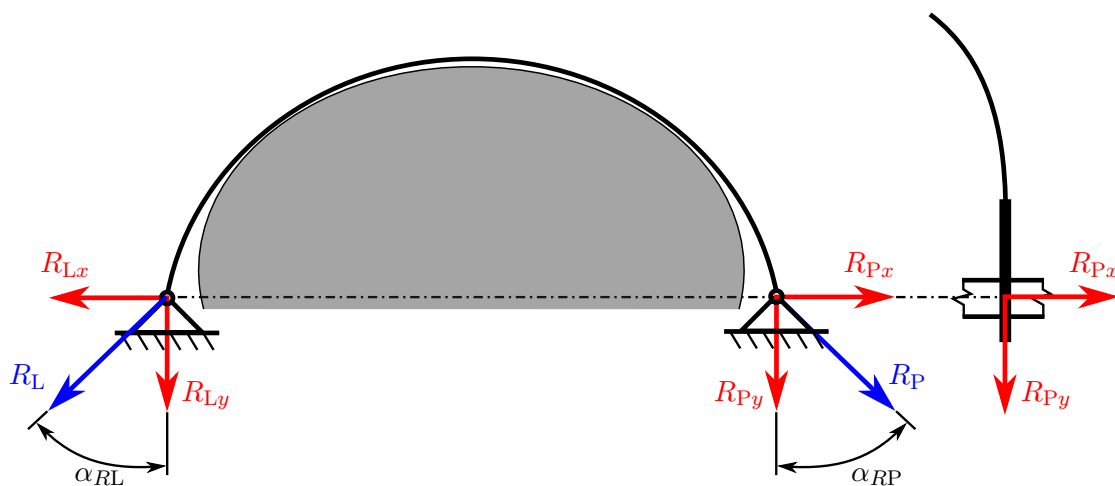
Ze srovnání je zřejmé, že průběhy se téměř shodují, což potvrzuje funkčnost vlastního algoritmu. Z průběhů reakcí je dále patrné, že v pravém kotevním bodě pásu vznikají při daných okrajových podmínkách znatelně větší reakční síly. Při změně směru natočení sedadla o úhel  $\alpha_2 = 10^\circ$  na druhou stranu by maximální reakce byla v levém kotevním bodě pásu. Maxima reakce je dosaženo v čase  $t_{\max} \approx 0,1$  s a její hodnota je  $R_{P\max} \approx 9650$  N. Maximální hodnota reakce v bodě L je  $R_{L\max} \approx 7093$  N. Pomocí jednotlivých složek lze určit směr působení výsledné reakce, který je důležitý při návrhu uložení pásu. Časové průběhy obou složek reakcí v obou bodech L a P jsou vykresleny na obr. 4.9 a 4.10.



Obrázek 4.8: Časový průběh výsledných reakcí v ukotvení pásu

Obrázek 4.9: Časový průběh složek reakce  $R_L$ 

Maximálních hodnot dosahují jednotlivé složky přibližně ve stejném okamžiku a jejich hodnoty jsou pro levou stranu  $R_{Lx} \approx -2466$  N a  $R_{Ly} \approx -6773$  N a pro pravou stranu  $R_{Px} \approx 4680$  N a  $R_{Py} \approx -8445$  N. Koncový díl pásu je zpravidla uložen na ocelovém čepu, který je působením reakcí namáhán na smyk, příp. na ohyb. Schématicky je působení reakcí v tomto čepu znázorněno na obr. 4.11.

Obrázek 4.10: Časový průběh složek reakce  $R_P$ 

Obrázek 4.11: Zatížení čepu

Úhly  $\alpha_{RL}$  a  $\alpha_{RP}$  se dopočtou ze vztahů

$$\alpha_{RL} = \arctan \left( \frac{R_{Lx}}{R_{Ly}} \right) \quad (4.12)$$

a

$$\alpha_{RP} = \arctan \left( \frac{R_{Px}}{R_{Py}} \right) \quad (4.13)$$

Výsledné hodnoty jsou  $\alpha_{RL} \approx 20^\circ$  a  $\alpha_{RP} \approx 29^\circ$ . Tyto hodnoty odpovídají daným okrajovým podmínkám, resp. směru zatížení. V případě otočení směru zatížení by byly hodnoty reakcí a jejich směrů pro levou a pravou stranu vzájemně zaměněny.

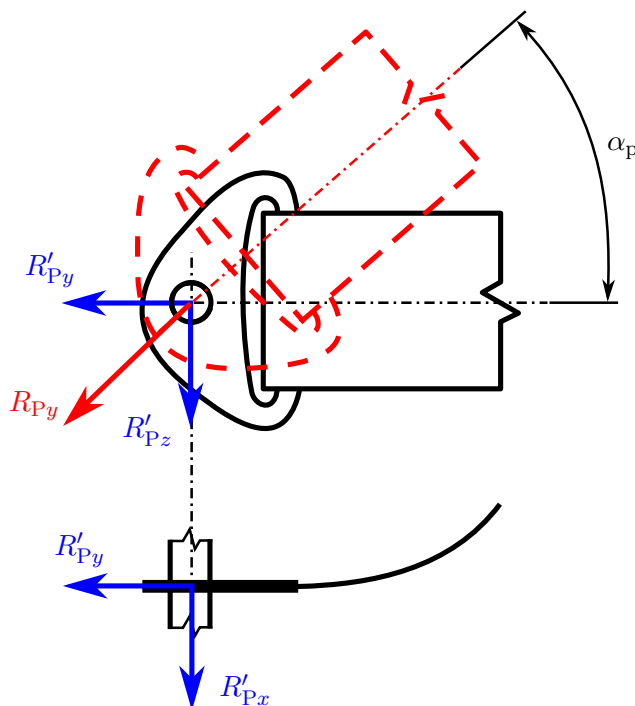
Lze tedy vyvodit závěr, že hodnota reakcí v obou bodech se pohybuje v rozmezí  $R \approx 7100 - 9700 \text{ N}$  a směr působení  $\alpha_R \approx 20 - 29^\circ$ .

Výše uvedené výsledky platí pro rovinnou úlohu, tedy pás ve vodorovné poloze. V případě natočení pásu v rozmezí  $\alpha_p$  dle kapitoly 2.1 se budou reakce lišit. Přibude složka reakce v ose z, jak je znázorněno na obr. 4.12 pro pravou stranu. Zjednodušeně lze předpokládat, že reakce v ose pásu je původní složka  $R_{Py}$ . Ta se při natočení pásu rozloží do složek  $R'_{Py}$  a  $R'_{Pz}$ , viz obr. 4.12. Zároveň dojde ke změně rozložení sil v rovině x-y, přičemž reakce v ose x zůstane nezměněná, tedy  $R'_{Px} = R_{Px}$ . Hodnoty těchto složek lze dopočíst ze vztahů

$$R'_{Py} = R_{Py} \cos(\alpha_p) \quad (4.14)$$

$$R'_{Pz} = R_{Py} \sin(\alpha_p) \quad (4.15)$$

$$R'_{Px} = R_{Px} \quad (4.16)$$



Obrázek 4.12: Zatížení čepu II

Obdobné vztahy a předpoklady platí analogicky pro reakce na levé straně v bodě L. Pro rozsah natočení střednice pásu  $\alpha_p = 30 - 55^\circ$  se pohybují výsledné hodnoty těchto složek v rozsahu  $R'_{Ly} \approx 5870 - 3890 \text{ N}$ ,  $R'_{Lz} \approx 3390 - 5550 \text{ N}$  pro levou stranu a pro pravou stranu  $R'_{Py} \approx 7310 - 4850 \text{ N}$ ,  $R'_{Pz} \approx 4220 - 6920 \text{ N}$ .

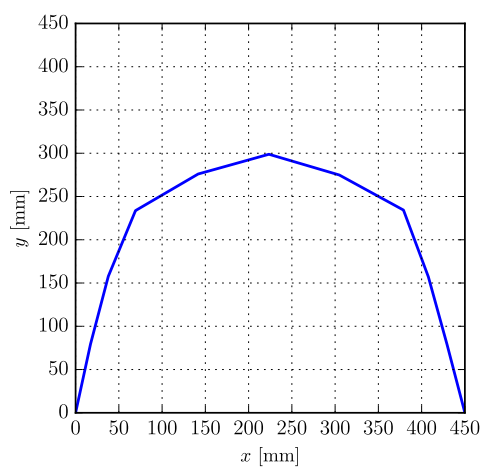
Na obr. 4.13 je dále vykreslena deformovaná geometrie pásu v několika časových okamžicích.



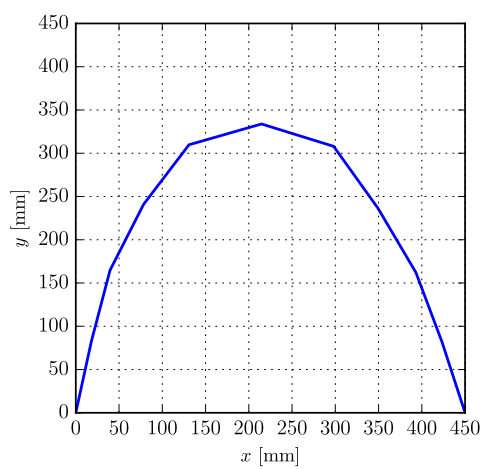
Je třeba mít na paměti, že při definici úlohy byla přijata řada zjednodušení. Následkem těchto zjednodušení dochází např. k deformaci pásu v průběhu testu. V reálném případě, nebo při modelování ve 3D pomocí MKP softwaru s figurínou, je výraznější deformaci geometrie pásu zabráněno právě kontaktem pásu s trupem figuríny. V tomto zjednodušeném případě však není deformaci nijak zabráněno, jak je patrné z průběhu deformace pásu během simulace na obr. 4.13.

Dalším zjednodušením je rozdělení hmotnosti figuríny přímo do uzlů diskretizovaného pásu, což si lze představit, jakoby byla figurína s pásem pevně spojena. V reálném případě však bezpečnostní pás nemusí být zcela utažený, či ve správné poloze a mezi pásem a sedící osobou může být mezera. Při fyzickém testu tak vznikne ráz od nárazu figuríny do pásu. Navíc dochází k dalším rázům při trhnutí končetin figuríny, což zde není zohledněno.

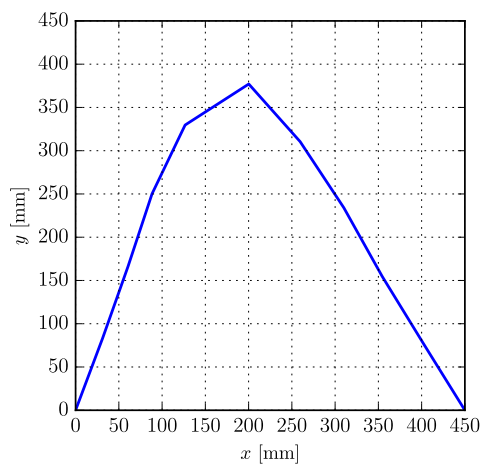
Přes všechny zjednodušení je však podoba průběhu reakcí s fyzickým testem uvedeným v [4] výborná, viz závěr.



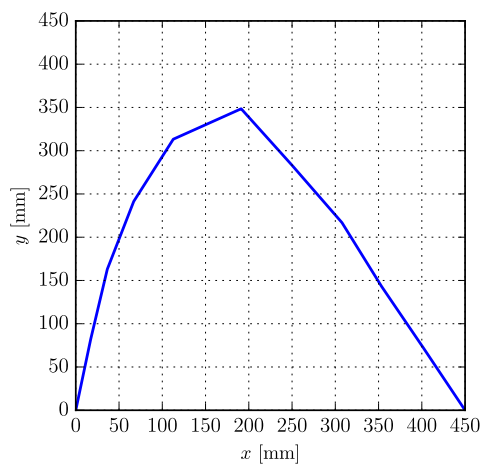
(a) 30 ms



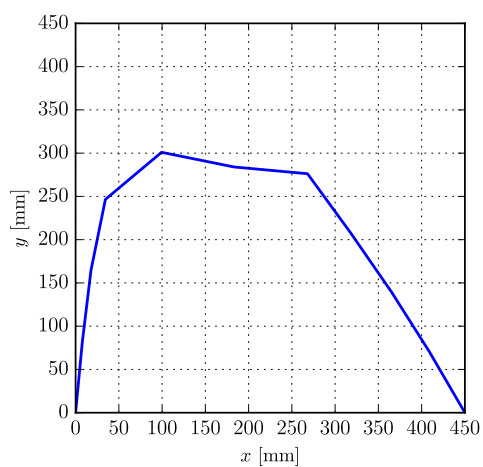
(b) 60 ms



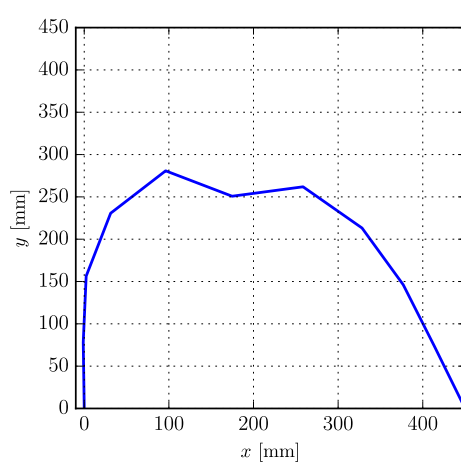
(c) 90 ms



(d) 120 ms



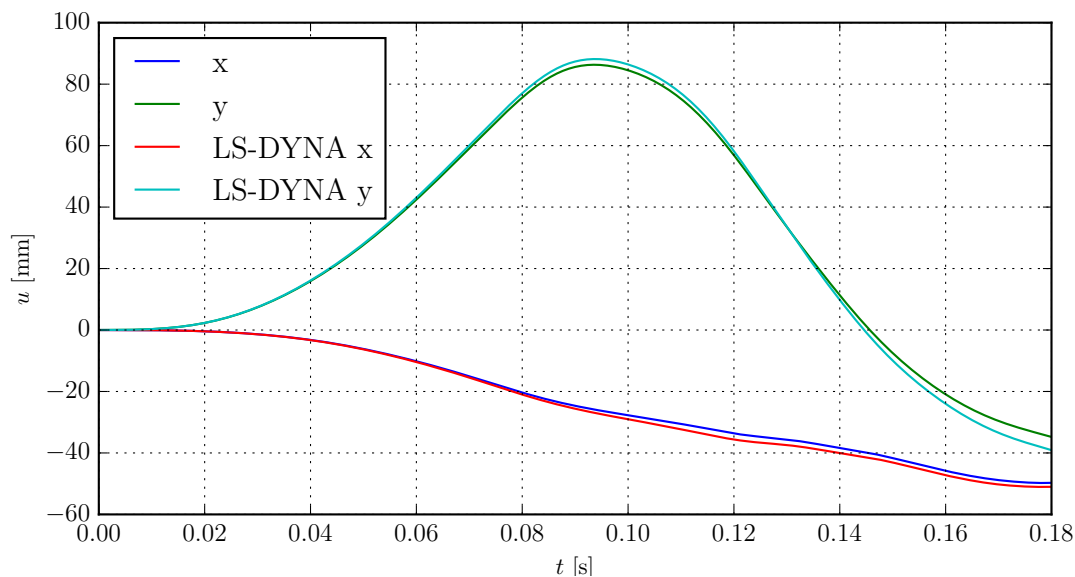
(e) 150 ms



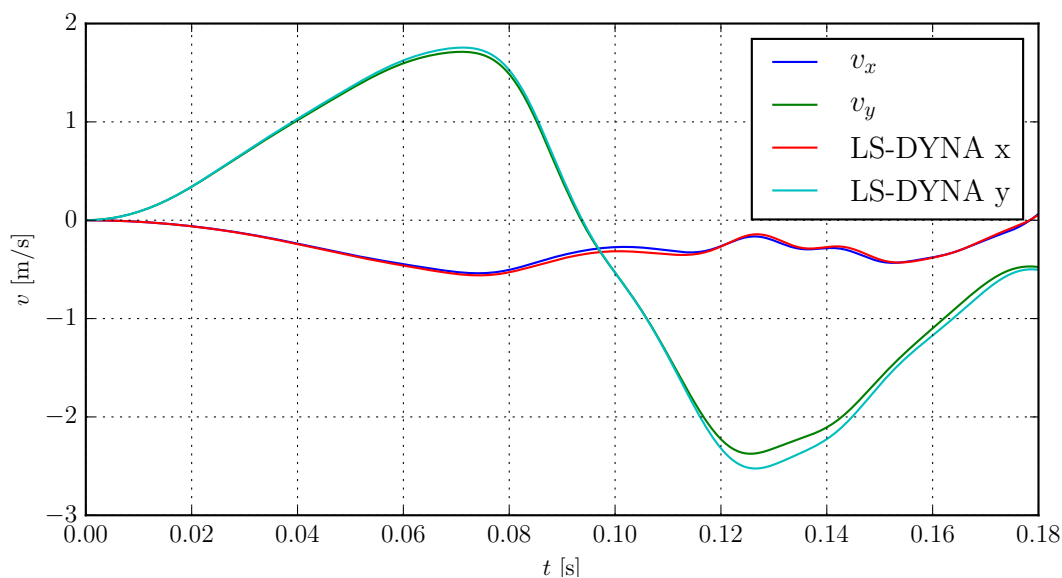
(f) 180 ms

Obrázek 4.13: Průběh deformace geometrie pásu

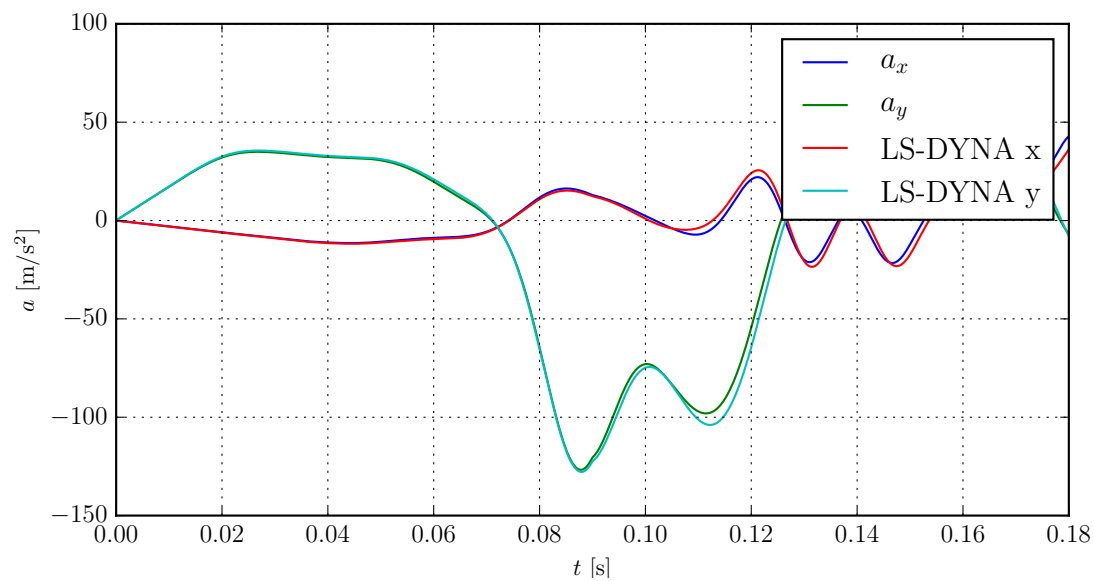
Na obr. 4.14 až 4.16 jsou pro zajímavost uvedeny časové průběhy výchylek, rychlostí a zrychlení středu pásu, získaných vlastním algoritmem a analýzou v LS-DYNA.



Obrázek 4.14: Časový průběh výchylky středu pásu



Obrázek 4.15: Časový průběh rychlosti středu pásu



Obrázek 4.16: Časový průběh zrychlení středu pásu

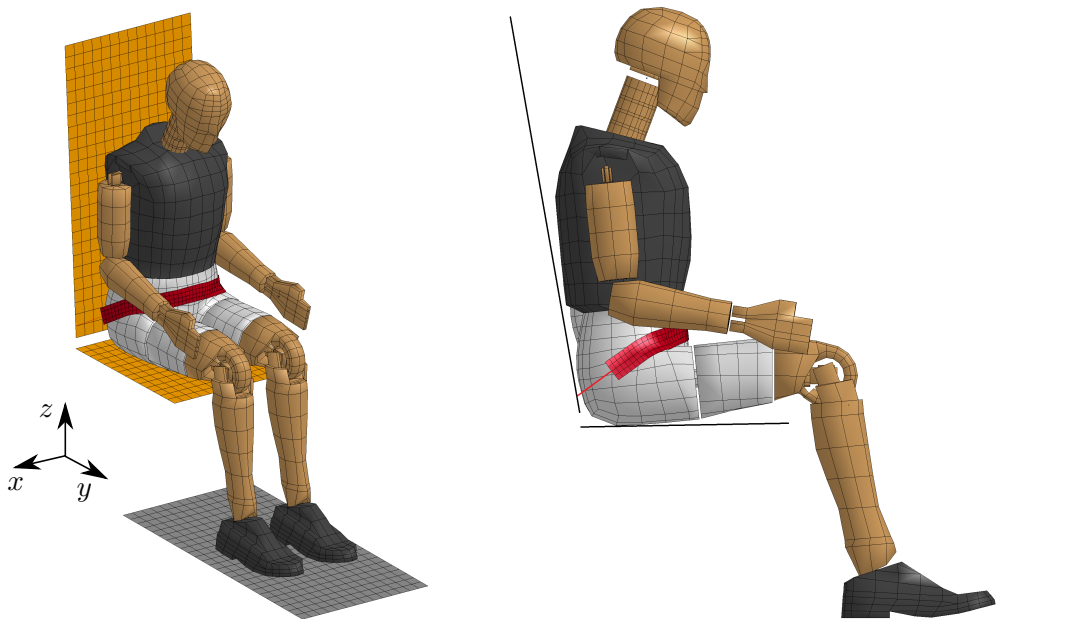
Vytvořený skript v jazyce Python, který byl použit k analýze, je k nalezení v příloze E. V další kapitole bude test bezpečnostního pásu modelován v softwaru LS-DYNA.

## 4.2 Analýza bezpečnostního pásu pomocí LS-DYNA

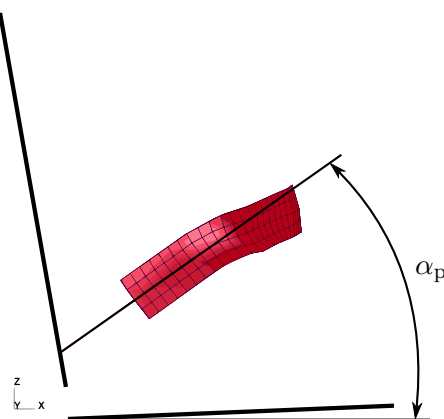
Analýza pomocí komerčního softwaru je provedena na prostorovém modelu, uvedeném v předchozí kapitole, který bude dále v textu popsán podrobněji. Podrobnější popis použitých definic modelu analýzy a popis použitých materiálových modelů je k nalezení v [15] a [16].

### 4.2.1 Popis modelu

V této analýze je využit výchozí 3D model uvedený v kapitole 4. Tento základní model je doplněný o tzv. kombinovaný bezpečnostní pás. Ten se skládá z plochy představující geometrii skutečného pásu a dvou křivek, které spojují tuto plochu a sedadlo. Celý model je na obr. 4.17. Poloha pásu a jeho ukotvení bylo zvoleno tak, aby byla splněna podmínka sklonu střednice pásu od vodorovné roviny, uvedená v kapitole 2.1 a jeho hodnota je  $\alpha_p = 31^\circ$ , viz obr. 4.18. Šířka a tloušťka membrány odpovídá rozměrům uvedeným v kapitole 2.3.



Obrázek 4.17: 3D výpočetní model s kombinovaným modelem pásu



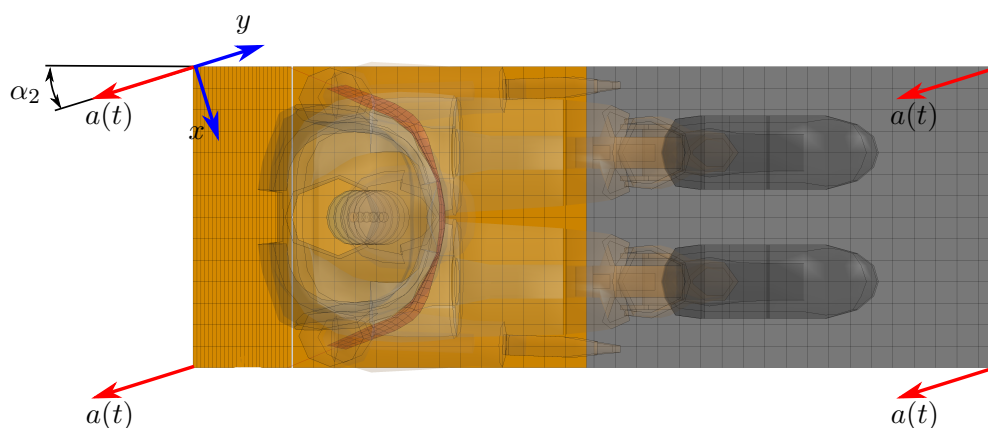
Obrázek 4.18: Sklon pásu

#### 4.2.2 Model sedadla a podlahy

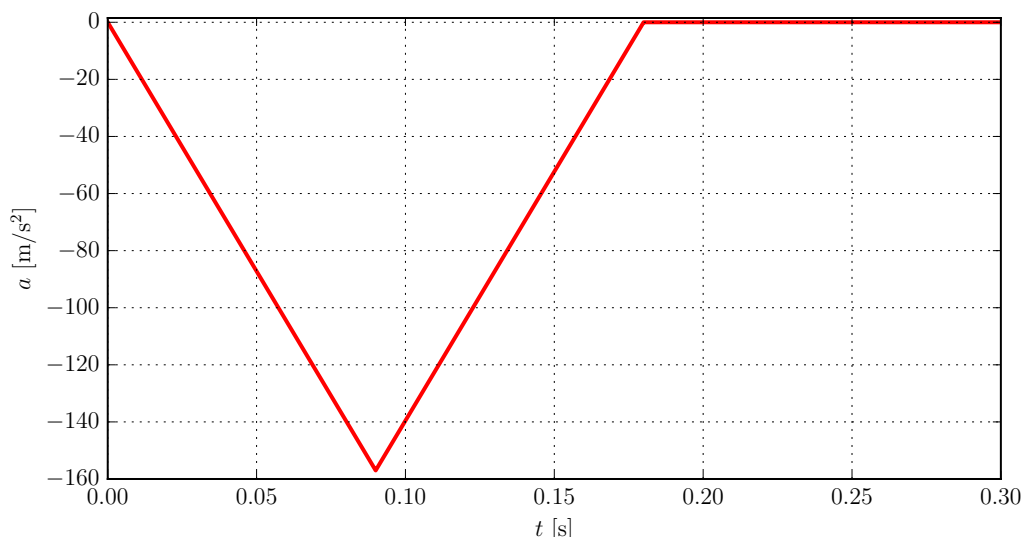
Sedadlo a podlaha jsou diskretizovány 4 uzlovými skořepinovými prvky s délkou hrany  $L_h \approx 32\text{mm}$ . Formulace elementu odpovídá plně integrovaným skořepinovým prvkům, jejichž analýza je velmi rychlá. Parametr tloušťky elementů je definován  $s_s = 1\text{mm}$ .

Sedadlu a podlaze je přiřazen absolutně tuhý materiál RIGID s materiálovými parametry oceli, tj. modul pružnosti  $E = 210\text{GPa}$ , Poissonovo číslo  $\mu = 0,3$  a hustota  $\rho = 7850\text{kg m}^{-3}$ . Díky absolutně tuhému materiálu a daným okrajovým podmínkám však konkrétní hodnoty nemají na analýzu vliv, mají však vliv na výpočet kontaktní tuhosti.

V definici materiálu RIGID je možné zadat rovněž okrajové podmínky v podobě vazeb. U této analýzy je požadovaná okrajová podmínka zrychlení, dle požadavků [19], zadána právě na sedadlo a podlahu. Znamená to, že sedadlo se pohybuje směrem dozadu a přes pás, který je k sedadlu uchycen, působí na figurínu. Takto zadaná okrajová podmínka vykazuje stejné výsledky, jako kdyby se sedadlo pohybovalo i s figurínou vpřed a poté byl zpomalením simulován náraz, jako je tomu u fyzického testu. Aplikace okrajové podmínky je schématicky znázorněna na obr. 4.19 a její časový průběh je vykreslen na obr. 4.20



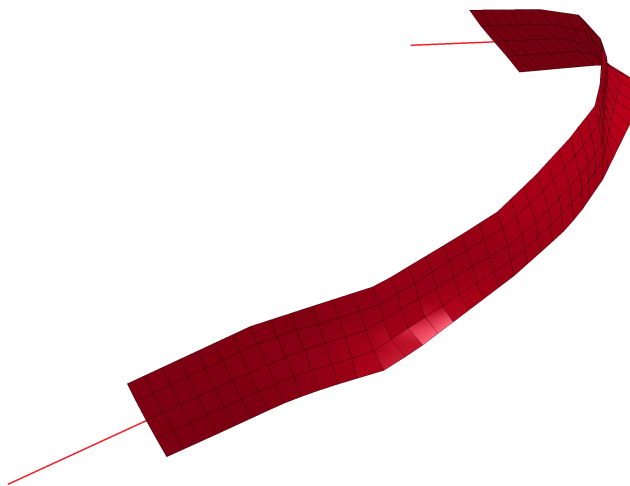
Obrázek 4.19: Definice okrajové podmínky



Obrázek 4.20: Časový průběh zrychlení sedadla

### 4.2.3 Model pásu

Diskretizovaný model pásu na obr. 4.21 je tvořen dvěma typy prvků. Část, která je v kontaktu s figurínou je tvořena 4 uzlovými membránovými prvky, které mají v každém uzlu pouze 3 stupně volnosti - posuvy ve třech na sebe kolmých osách (tento typ elementů tedy nepřenáší ohybové namáhání). Délka hrany prvku je  $L_h \approx 12\text{ mm}$ . Na šířku pásu  $b_p = 48\text{ mm}$  připadají čtyři prvky. Tloušťka elementů je  $s_p = 1,2\text{ mm}$ .



Obrázek 4.21: Diskretizovaný model pásu

Membránové části pásu je přiřazen materiál s označením FABRIC. Tento materiál se

používá k modelování airbagů, ale je možné jej použít i pro modelování bezpečnostních pásů. Jedná se o vrstvený ortotropní kompozitní materiálový model a je platný pouze pro 3 a 4 uzlové membránové elementy. Umožňuje zadání mnoha parametrů materiálu, mezi které patří:

- hustota
- orientace vláken materiálu
- modul pružnosti, Poissonovo číslo a smykový modul ve třech na sebe kolmých směrech, dle orientace vláken
- eliminace tlakového napětí v materiálu
- koeficient Rayleighova tlumení

Zde jsou za předpokladu izotropního materiálu hodnoty modulů pružnosti, Poissonových čísel a smykových modulů ve všech směrech stejné a odpovídají hodnotám uvedeným v tab. 2.3.

Při aktivované eliminaci tlakového napětí nemají elementy žádnou tuhost v tlaku, což může mít za následek dokonce zhroucení prvku do jedné linie a přerušení výpočtu, viz [15]. Při této analýze se tato volba neosvědčila.

Posledním uvedeným parametrem je koeficient tlumení, který má vliv na stabilitu výpočtu. Dle [16] je doporučená hodnota 0,05, což odpovídá 5 % kritického tlumení. Manuál však zároveň uvádí, že v některých případech je nutná hodnota větší. V této úloze bylo uvažováno tlumení s koeficientem 0,2.

Důležitým prvkem v definici materiálu je materiálová křivka získaná z tahové zkoušky pásu. V tomto případě byla použita linearizovaná křivka, uvedená v kapitole 2.2, pro nylon. Zadávají se zde dvě křivky, a to pro podélný a pro příčný směr pásu. Tyto směry je nutné správně definovat. Pro podélný směr je použita materiálová křivka v nezměněné podobě. V příčném směru je typicky závislost síly na deformaci pásu poloviční.

Druhá část pásu, tj. spojení membránové části pásu s opěradlem sedadla, je realizováno pomocí speciálních prvků SEATBELT. Těmto prvkům je nutné přiřadit pouze materiál, který má označení SEATBELT. V definici materiálu je v tomto případě zadána pouze hodnota hmotnosti pásu na jednotku délky o velikosti  $\rho_m = 0,08 \text{ kg m}^{-1}$  a přiřazena materiálová křivka pro zatížení a odlehčení z kapitoly 2.2. Spojení těchto jednorozměrných prvků s membránovou částí pásu je realizováno pomocí vazeb, které spojí uzly na konci membránové části s prvním uzlem liniové části.

#### 4.2.4 Model figuríny

Model testovací figuríny je poskytován plně definovaný a zkalibrovaný, jak již bylo uvedeno dříve. K diskretizaci figuríny je využito řady prvků, od bodových elementů až po objemové, a řada materiálů. Jsou zde rovněž přednastaveny různé senzory, tj. uzly, ve kterých se v průběhu výpočtu zaznamenávají výstupní veličiny, jako rychlosti, zrychlení, silové nebo momentové reakce apod. Na model figuríny působí během výpočtu tíhové zrychlení.



#### 4.2.5 Definice kontaktů

Definice kontaktů má velký vliv na celou analýzu. Jejich nesprávné nastavení může vést ke zcela nereálným výsledkům. Z toho důvodu se zpravidla provádí ladění nastavení kontaktů a celé analýzy podle výsledků reálného fyzického testu. Výsledky reálného testu leteckého sedadla dle [19] jsou k nalezení v [4].

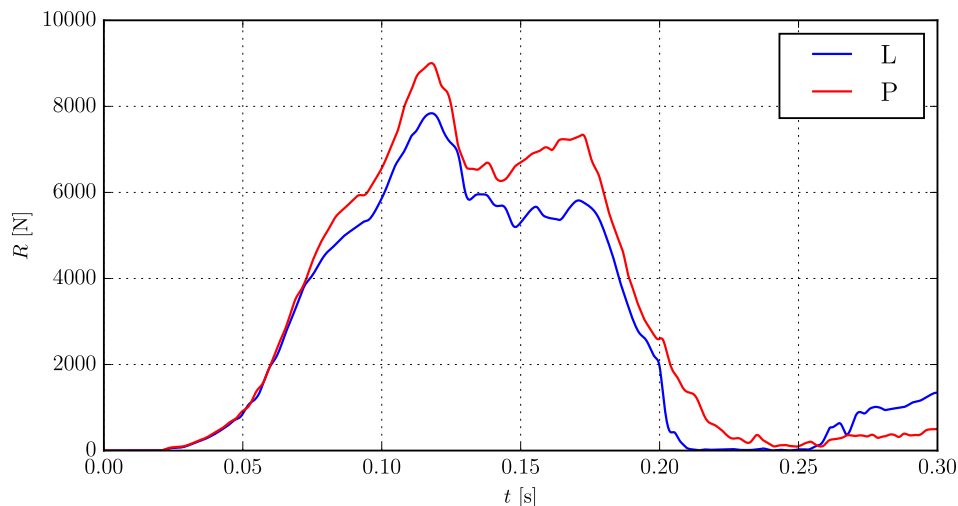
Řada kontaktů je již přednastavena v modelu figuríny. Přednastavené kontakty byly doplněny o chybějící kontakty mezi některými částmi těla figuríny a parametry těchto kontaktů byly převzaty z přednastaveného modelu.

Další kontakty jsou předepsány mezi figurínou a podlahou, figurínou a sedadlem a mezi figurínou a bezpečnostním pásem. V nastavení kontaktu je možné definovat statický a dynamický součinitel smykového tření. Hodnoty těchto koeficientů byly odborným odhadem určeny na hodnotu 0,8.

Posledním použitým nastavením je definice algoritmu pro detekci kontaktu. Jak bylo uvedeno, standardně je nastaven penaltní přístup, který v každém kroce testuje penetraci uzlu kontaktního tělesa s elementem kontaktovaného tělesa. Při kontaktu těles s odlišnou tuhostí materiálu (např. ocel-pěna) je však vhodnější použít algoritmus označený jako Segment-based Penalty Formulation, který vyhodnocuje penetraci mezi elementem kontaktního a elementem kontaktovaného tělesa.

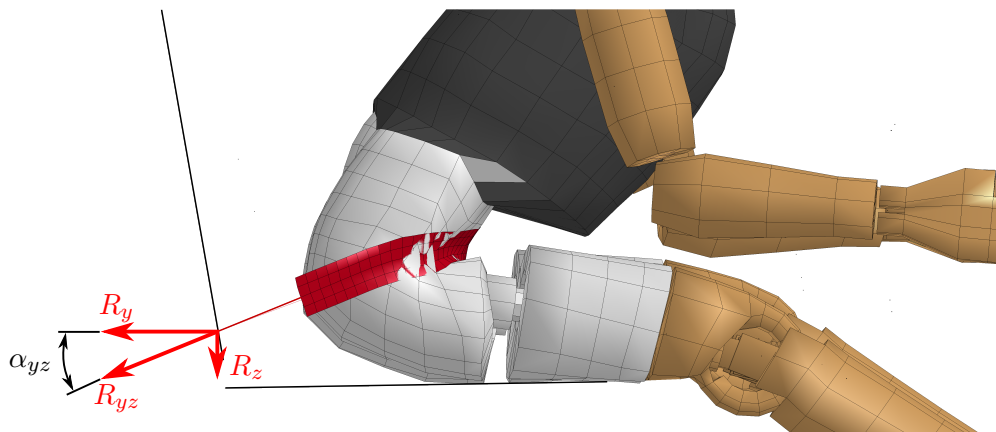
#### 4.2.6 Výsledky analýzy

Primárními výsledky této analýzy jsou opět průběhy reakcí v pásu, resp. v jeho kotevních bodech. Získané průběhy jsou vykresleny na obr. 4.22 (průběhy jsou filtrovány filtrem s označením SAE 108 Hz).



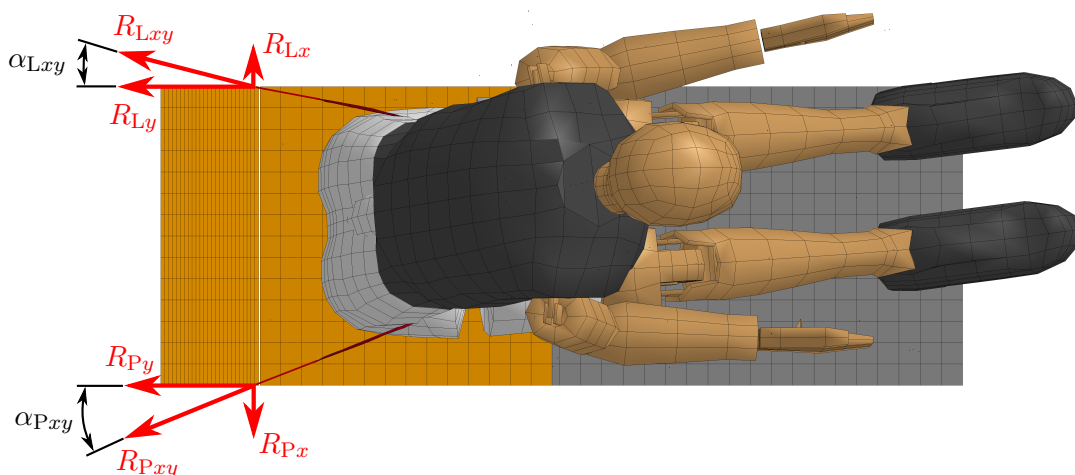
Obrázek 4.22: Reakce v kotevních bodech pásu

Značení L a P je stejné jako v předchozí úloze a je zřejmé, že větší hodnoty vykazuje dle očekávání reakce na pravé straně  $R_P$ . Maximální hodnota reakce zde dosahuje  $R_{P\max} \approx 9000 \text{ N}$ . Směr působení reakcí lze určit z geometrie v průběhu testu. Na obr. 4.23, 4.24 a 4.25 jsou naznačeny směry reakcí a jejich složek v čase  $t = 118 \text{ ms}$ , kdy jsou reakce maximální.

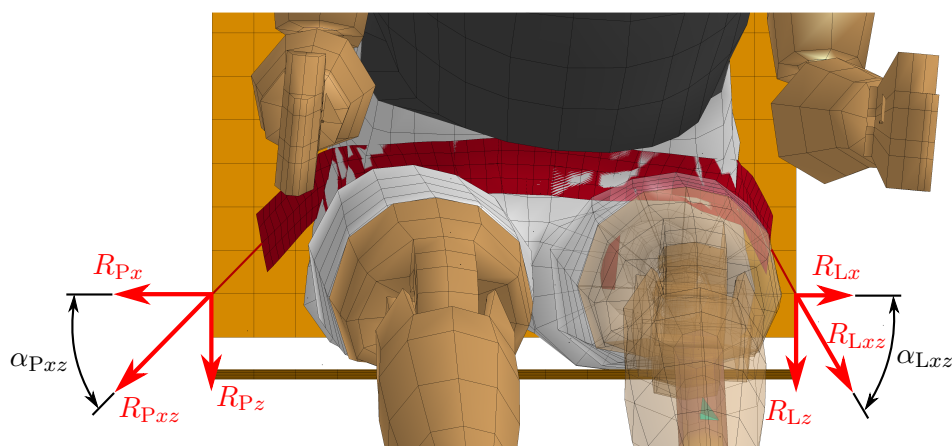


Obrázek 4.23: Směr působení reakcí

Hodnoty vyznačených úhlů jsou  $\alpha_{Lxy} \approx 12^\circ$ ,  $\alpha_{Pxy} \approx 22^\circ$ ,  $\alpha_{Lxz} \approx 61^\circ$  a  $\alpha_{Pxz} \approx 45^\circ$ . Úhel  $\alpha_{yz}$  je pro obě strany přibližně stejný a má hodnotu  $\alpha_{yz} \approx 22^\circ$ .



Obrázek 4.24: Směr působení reakcí II



Obrázek 4.25: Směr působení reakcí III

Z analýzy jsou známy výsledné reakce v pásu  $R_L$  a  $R_P$ , které lze zapsat jako

$$R_L = \sqrt{R_{Lx}^2 + R_{Ly}^2 + R_{Lz}^2} \quad (4.17)$$

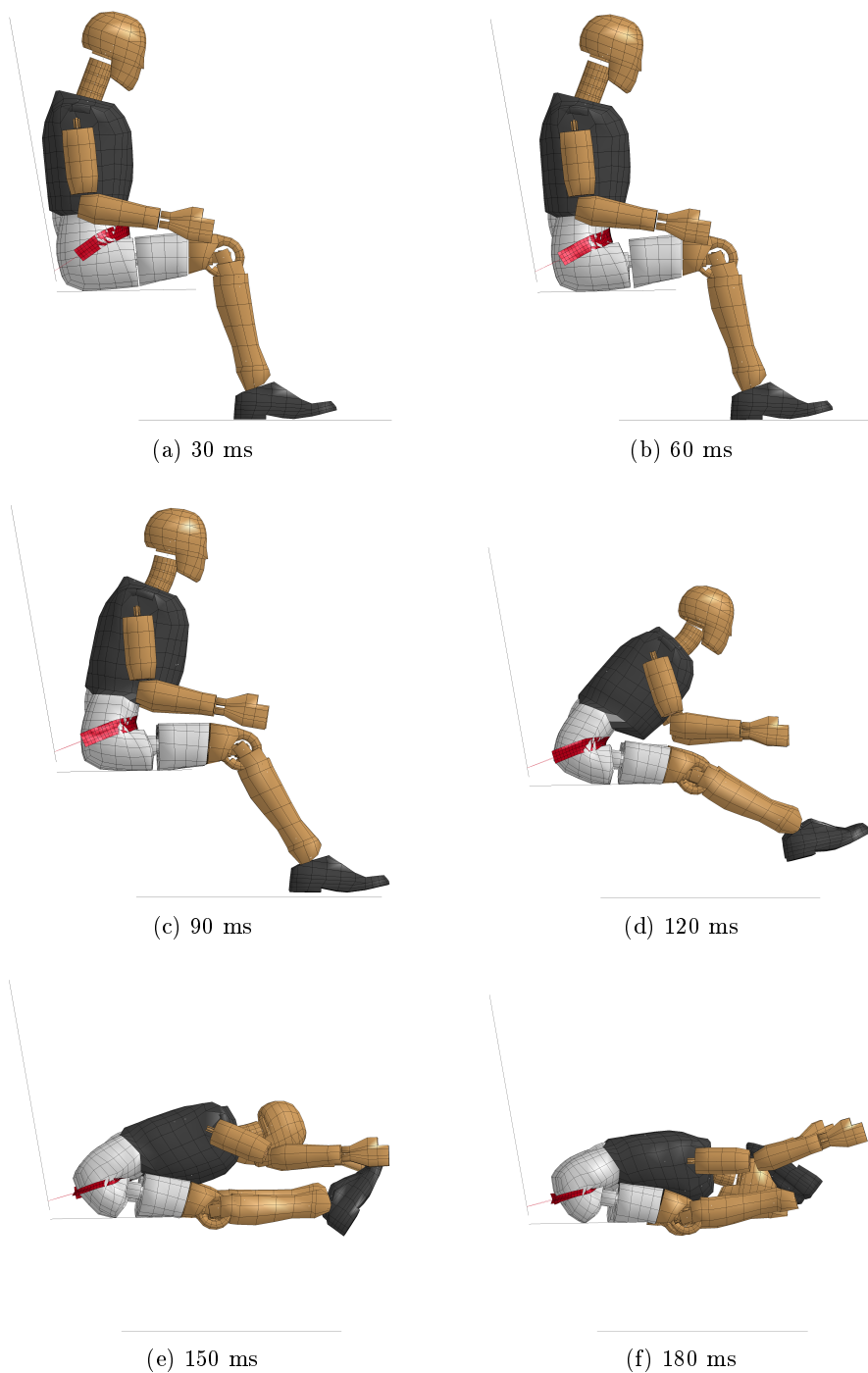
a

$$R_P = \sqrt{R_{Px}^2 + R_{Py}^2 + R_{Pz}^2}. \quad (4.18)$$

Pomocí směrových kosinů výsledných reakcí ke globálním souřadným osám lze dopočítat velikost jednotlivých složek reakcí. Jejich hodnoty jsou pro levou stranu  $R_{Lx} \approx 1460 \text{ N}$ ,  $R_{Ly} \approx 7220 \text{ N}$  a  $R_{Lz} \approx 2700 \text{ N}$  a pro pravou stranu  $R_{Px} \approx 3130 \text{ N}$ ,  $R_{Py} \approx 7820 \text{ N}$  a  $R_{Pz} \approx 3210 \text{ N}$ . Dílčí výslednice v jednotlivých rovinách  $R_{xy}$ ,  $R_{yz}$  a  $R_{xz}$  lze dopočítat pomocí Pythagorovy věty nebo s využitím goniometrických funkcí.

Působení těchto reakcí v čepu je obdobné jako v kapitole 4.1.4. Rovněž zde platí, že při změně směru působení zrychlení budou reakce stejné, jen vzájemně zaměněny.

Dalším výsledkem analýzy jsou vizualizace průběhu testu. Na obr. 4.26 je zobrazeno několik stavů v časových okamžicích po 30 ms a v příloze F je přiložen průběh simulace z různých pohledů ve formě animace.



Obrázek 4.26: Průběh simulace

## Kapitola 5

# Závěr a zhodnocení

Tato diplomová práce vznikla na základě požadavků praxe společnosti IDEA AIR s.r.o., která se zabývá vývojem leteckých sedadel. Cílem práce bylo stanovení časových průběhů reakcí a jejich maximálních hodnot v kotevních bodech bezpečnostního pásu při nárazové zkoušce leteckého sedadla.

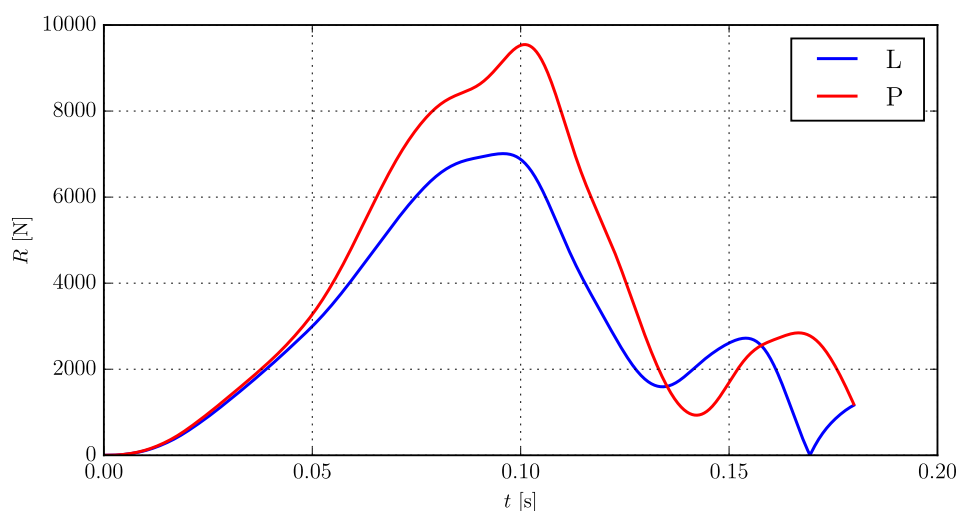
V první části je provedena rešerše problematiky testování leteckých sedadel, kterou popisuje mezinárodní norma pro letectví **SAE AS8049C**. Dále byly stanoveny rozměry bezpečnostního pásu a materiálové konstanty použitého materiálu.

Další část se zabývá matematickým pozadím modelování dynamických úloh pomocí MKP. Je proveden rozbor a odvození vztahů pro explicitní schéma řešení diskretizovaných pohybových rovnic. Pro diskretizaci kontinua je zde využit rovinný tyčový prvek. Odvozené vztahy jsou dále aplikovány na několika příkladech a implementovány v programovacím jazyce Python, s cílem vytvořit vlastní výpočetní algoritmus pro řešení dynamických úloh. Řešení pomocí vlastního algoritmu je vždy srovnáno se známým analytickým řešením nebo s komerčním softwarem. Verifikovaný kód je následně využit k analýze bezpečnostního pásu pomocí MKP ve 2D. Výsledkem jsou časové průběhy reakcí v kotevních bodech pásu.

V poslední části práce je provedena analýza bezpečnostního pásu pomocí MKP ve 3D s využitím komerčního softwaru LS-DYNA. Výsledkem jsou opět časové průběhy reakcí a vizualizace simulace testu sedadla.

## 5.1 Zjednodušený 2D model

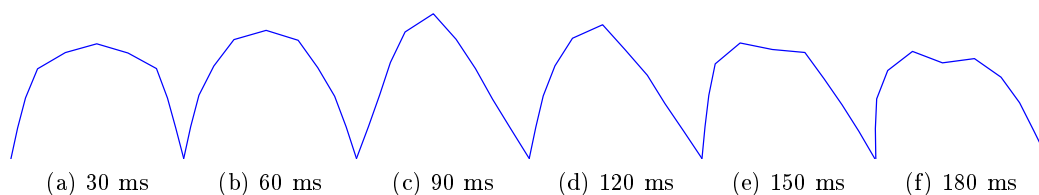
Pro rovinnou analýzu bezpečnostního pásu byl využit vlastní skript vytvořený v programovacím jazyce Python. Geometrii tvoří střednice pásu odvozená z prostorového modelu. Křivka je diskretizována 10 rovinnými tyčovými elementy, jejichž konečnoprvková formulace je odvozená v kapitole 3.2.2. Zadání okrajových podmínek je zřejmé z obr. 4.7. Krajní uzly diskretizovaného pásu jsou kloubově uloženy a zbylým uzlům je předepsáno zrychlení s požadovaným časovým průběhem dle [19]. Výsledné časové průběhy v kotevních bodech jsou vykresleny na obr. 5.1.



Obrázek 5.1: Reakce v kotevních bodech pásu - 2D analýza

Analýzou byla zjištěna maximální hodnota reakce v pravém kotevním bodě odpovídající  $R \approx 9650 \text{ N}$ . Tento výsledek platí pro dané okrajové podmínky, zejména pro natočení sedadla o úhel  $\alpha_2 = 10^\circ$ , viz obr. 4.7. Pro opačné natočení by uvedená hodnota reakce byla v levém kotevním bodě pásu. Orientační směry působení reakcí jsou vykresleny na obr. 4.11 a 4.12.

Při řešení úlohy byla přijata řada zjednodušujících předpokladů. Především je zde uvažován materiál pásu s lineárním chováním při zatížení v tahu. Dalším zjednodušením je sklopení střednice pásu do roviny a rozložení hmotnosti figuríny přímo do uzlů diskretizovaného pásu. Na obr. 5.2 je patrný jeden z důsledků těchto přijatých zjednodušení. Na obrázku

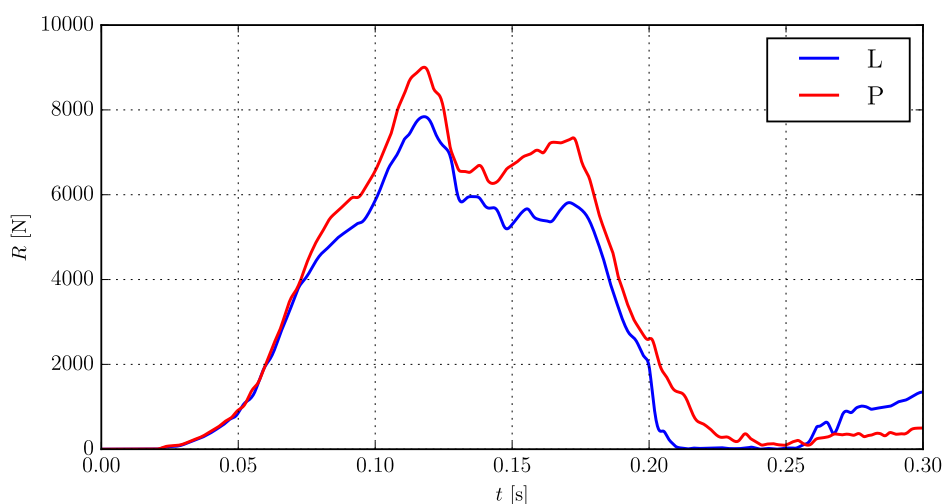


Obrázek 5.2: Průběh deformace geometrie pásu

je vykreslena deformovaná geometrie pásu v průběhu analýzy. Při reálném experimentu je však podobné deformaci zabráněno kontaktem s testovací figurínou, kterou pás obepíná. Přes to je však časový průběh reakcí velice blízký výsledkům fyzického testu uvedeného v [4].

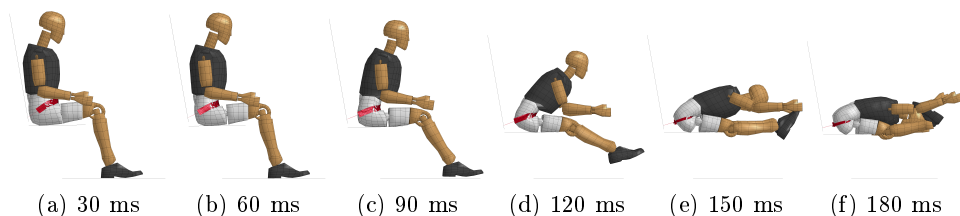
## 5.2 3D model

Pro prostorovou analýzu bezpečnostního pásu byl využit komerční software LS-DYNA. Model úlohy je tvořen zjednodušeným sedadlem z absolutně tuhého materiálu, modelem pásu a modelem testovací figuríny. Zadání okrajových podmínek je zřejmé z obr. 4.19. Požadované zrychlení je zde předepsáno sedadlu, které je v kontaktu s figurínou a zároveň je uvažován vliv gravitace. Výsledné časové průběhy v kotevních bodech jsou vykresleny na obr. 5.3.



Obrázek 5.3: Reakce v kotevních bodech pásu - 3D analýza

Analýzou byla zjištěna maximální hodnota reakce v pravém kotevním bodě odpovídající  $R \approx 9000 \text{ N}$ . Orientační směry působení reakcí jsou vykresleny na obr. 4.23, 4.24 a 4.25. Na obr. 5.4 je vizualizace průběhu simulace.



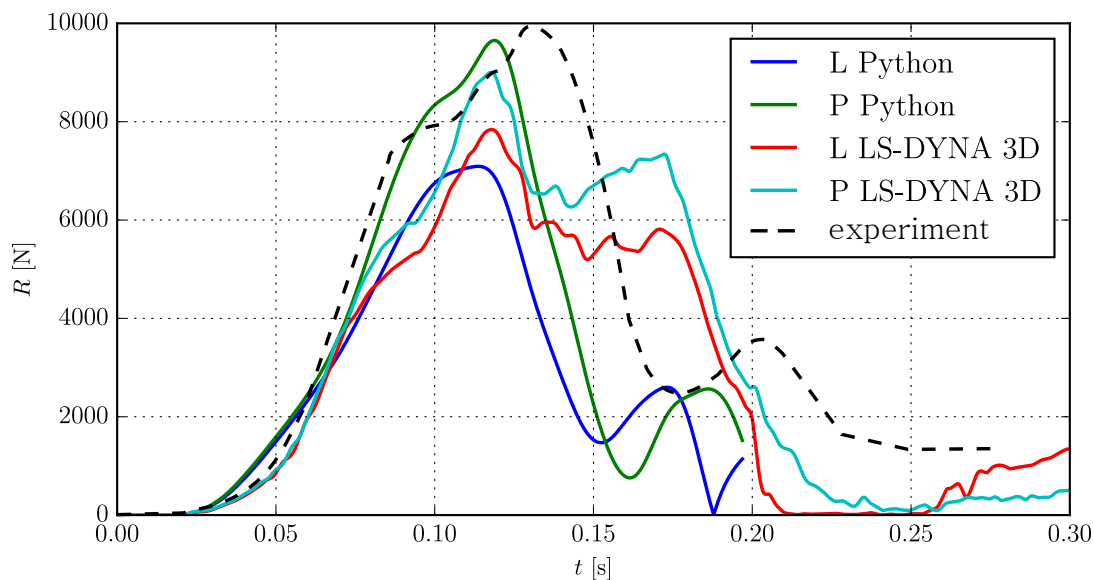
Obrázek 5.4: Průběh simulace

Při této analýze bylo přijato rovněž několik zjednodušení. Sedadlo je v ranné fázi vývoje a jeho podrobná geometrie v současné době není známa. Z toho důvodu tvoří model sedadla pouze dvě absolutně tuhé desky. Stejně tak není známa poloha kotevních bodů bezpečnostního pásu na sedadle. Pro návrh uložení a stanovení orientačních hodnot je však model dostatečně přesný.

### 5.3 Zhodnocení a doporučení

Ze srovnání časových průběhů reakcí na obr. 5.1 a 5.3 se jeví průběhy z LS-DYNA oproti průběhům z 2D analýzy jako zpožděné. Tento posun je dán především poddajností konečnoprvkového modelu figuríny. Na počátku působení zrychlení dojde ke stlačení těla figuríny s minimálním odporem, což je patrné i z obr. 4.26. Po určitém stlačení těla figuríny odpor vzroste a dále již jsou průběhy podobné. Dalším důvodem delšího náběhu je utažení a vedení pásu kolem těla figuríny. Mezi pásem a figurínou může být malá vůle, případně může pás mírně sklouznout, což se projeví delším náběhem s téměř nulovými reakcemi v kotevních bodech. Počátek průběhů reakcí získaných z vlastního algoritmu je proto pro vyrovnání náběhů posunut o  $\Delta t = 17$  ms.

Na obr. 5.5 jsou srovnány časové průběhy reakcí získané pomocí vlastního algoritmu, pomocí LS-DYNA a průběh reakcí z experimentu, který je proveden v [4]. U zmíněného experimentu nebylo zahrnuto natočení sedadla o úhel  $\alpha_2 = 10^\circ$  a reakce v obou kotevních bodech jsou prakticky totožné.



Obrázek 5.5: Srovnání průběhů reakcí v kotevních bodech bezpečnostního pásu

Z obr. 5.5 je patrné, že maximální reakce získaná experimentem a reakce získaná s pomocí vlastního algoritmu mají i přes veškerá zjednodušení velice podobné průběhy. Rovněž maximální hodnoty se příliš neliší.



Ve srovnání s průběhy z LS-DYNA už je však patrný větší rozdíl. Rozdíl ve výsledcích může být způsoben použitým modelem sedadla z absolutně tuhého materiálu. Tím je zabráněno jakékoliv deformaci sedadla a toto zjednodušení může mít nemalý vliv na výsledné hodnoty. Navíc není známa přesná poloha kotevních bodů pásu, která spolu se sklonem sedáku výrazně ovlivňuje velikost reakcí. Dalším důvodem může být použitá materiálová charakteristika bezpečnostního pásu. V této práci byla použita linearizovaná charakteristika odečtená z [4]. Významným faktorem je rovněž matematický popis kontaktních dvojic. Především pak kontaktů mezi figurínou a sedadlem a figurínou a bezpečnostním pásem, resp. nastavením součinitele smykového tření mezi těmito povrchy. Protože nejsou známy ani materiály, ze kterých bude sedadlo vyrobeno, byly tyto koeficienty určeny odborným odhadem. V neposlední řadě je zde možnost využít přesnějšího modelu testovací figuríny.

V tab. 5.1 jsou vypsané maximální hodnoty reakcí získaných uvedenými třemi způsoby.

	maximální síla $R$ [N]
<b>2D analýza</b>	9650
<b>3D analýza</b>	9000
<b>experiment</b>	10 000

Tabulka 5.1: Dosažené maximální hodnoty reakcí v kotevních bodech bezpečnostního pásu

Bezpečnost cestujících je vždy na prvním místě. Pro návrh kotvení bezpečnostního pásu v leteckých sedadlech lze proto doporučit uvážení vyšší hodnoty reakcí. Při zaokrouhlení hodnoty získané analýzou vlastním algoritmem je tedy doporučená hodnota reakce  $R \approx 10\,000$  N. To je také hodnota získaná experimentem. Směry působení reakcí uvedené v této práci je třeba brát spíše jako orientační. Přesné směry budou záviset na konkrétní poloze kotevních bodů bezpečnostního pásu.

## 5.4 Využití výsledků

Výsledky budou poskytnuty společnosti IDEA AIR s.r.o. pro návrh nového typu leteckého sedadla. Zároveň bude úloha modifikována dle případných připomínek společnosti.

Díky přiloženým vytvořeným skriptům může tato práce sloužit i k didaktickým účelům. Modifikací uvedených algoritmů lze rovněž řešit různé úlohy dynamiky s využitím rovinného tyčového prvku.

Práce bude také prezentována na konferenci 14 th International Conference of Numerical Analysis and Applied Mathematics ICNAM 2016 Rhodes, Greece, <http://icnaam.org>, a publikována v příslušném sborníku. Vytvořený algoritmus bude registrován formou aplikovaného výstupu (software), který bude řešit silovou odezvu na základě známého přetížení.

## 5.5 Pokračování práce

Ze srovnání průběhů z 2D analýzy, 3D analýzy a z experimentu je zřejmé, že i přes řadu zjednodušení dosahuje 2D analýza s využitím vlastního algoritmu výborných výsledků. Použité postupy budou dále rozšířeny pro prostorové úlohy a budou implementovány další typy prvků, pro možnost provádění dynamických analýz složitějších struktur. Výsledný algoritmus by mohl nalézt uplatnění v softwaru pro řešení rozsáhlých dynamických úloh, bez omezení licencemi komerčních produktů, např. na superpočítačích Národního superpočítačového centra IT4Innovations, v jehož spolupráci tato diplomová práce vznikla.

## Poděkování

Na tomto místě bych rád poděkoval vedoucímu práce Ing. Pavlu Maršálkovi za odborné vedení a věnovaný čas. Dále bych chtěl poděkovat Ing. Alexandru Markopoulosovi, Ph.D. a prof. Ing. Petru Horylovi, CSc., dr.h.c. za cenné rady při zpracování práce. Poděkování patří rovněž mé rodině a přátelům za jejich podporu.

Tato práce byla podpořena VŠB-TU Ostrava grantem SGS SP2016/176 a Ministerstvem školství mládeže a tělovýchovy z podpory Velkých infrastruktur pro výzkum, experimentální vývoj a inovace v rámci projektu „IT4Innovations národní superpočítačové centrum - LM2015070“.



## Příloha A

# Diskrétní systém

---

```
1 import numpy as np
2 import matplotlib as mpl
3 import matplotlib.pyplot as plt
4
5 #pozn. algoritmus je shodny pro tlumene i netlumene kmitani;
6 #volba zavisi pouze na hodnote soucinitele tlumeni b:
7 #netlumeny system: b=0; tlumeny system b>0
8
9 #netlumene/tlumene kmitani analyticky
10 def u_an(t):
11     return
12         np.exp(-delta*t)*(u0*np.cos(omega*t)+((v0+u0*delta)/omega)*np.sin(omega*t))
13
14 k = 1.      #tuhost pruziny
15 m = 1.      #hmotnost telesa
16 b = 0.0     #koeficient tlumeni
17 u0 = 1.     #pocatecni vychylka
18 v0 = 0.     #pocatecni rychlost
19 F = 0.      #vnejsi zatizeni
20
21 omega0 = k/m          #vlastni kruhova frekv. netlumeneho kmitani
22 delta = b/(2*m)       #konstanta doznavani
23 omega = np.sqrt(omega0**2-delta**2) #vl. kruh. frekvence kmitani
24 perioda = 2*np.pi/omega #perioda kmitani
25 T = 4*perioda         #delka analyzy
26
27 #vykresleni analytickeho reseni
28 mpl.rcParams['text.latex.unicode']=True #styl pisma LaTeX
29 mpl.rc('font',**{'family':'serif','serif':['Computer Modern Roman']})
30 mpl.rcParams['text.usetex']=True
31 Tt = np.linspace(0,T,100)
32 plt.figure(1,figsize=(8,4))
33 plt.plot(Tt,u_an(Tt),'--',hold=True,color="black",lw='2',label=u"analyticky")
34 plt.xlabel('$t$ [s]',fontsize=11)
35 plt.ylabel('$u$ [m]',fontsize=11)
36 plt.legend(loc='lower right')
37 plt.grid(True)
```

```

36 #explicitni algoritmus v kombinaci s promennou velikosti casoveho kroku
37 for i in range(2,6):
38     dt = 0.1*i*perioda/np.pi #promenna velikost casoveho kroku
39     n = int(T/dt)+1           #pocet vypocetnich kroku
40     #startovaci procedura
41     u_t1 = u0
42     u_t2 = 0.
43     du1 = v0
44     du2 = -k/m*u0             #pocatecni zrychleni
45     u_start = u0-dt*du1+dt**2/2*du2 #startovaci procedura
46     u_t0 = u_start
47     u_numt1 = np.zeros(n+1,dtype=np.float64)
48     u_numt1[0] += u0
49     T_num = np.zeros(n+1,dtype=np.float64)
50     T_num[0] += 0.
51     T_total= dt
52     j = 1
53     #explicitni algoritmus
54     while (T_total < T+dt) or (T_total == T+dt):
55         u_t2 = 1/(m/dt**2+b/(2*dt))*(F-(k-2*m/dt**2)*u_t1-(m/dt**2-b/(2*dt))*u_t0)
56         u_numt1[j] += u_t2
57         T_num[j] += T_total
58         print(T_total, '', u_t2)
59         T_total += dt
60         j += 1
61         #ulozeni vysledku
62         u_t0 = u_t1
63         u_t1 = u_t2
64         u_t2 = 0.
65     #vykresleni numerickeho reseni
66     plt.figure(1,figsize=(8,4))
67     plt.plot(T_num,u_numt1,'.-',hold=True,label='$\eta=%s'% (dt))
68     plt.xlabel('$t$ [s]',fontsize=11)
69     plt.ylabel('$u$ [m]',fontsize=11)
70     plt.legend(loc='lower right')
71     plt.grid(True)
72     plt.show()

```

---

## Příloha B

# Struna

---

```
1 import numpy as np
2 from scipy import linalg as la
3
4 #generator site
5 def getMesh(L,nEl):
6     nNod=nEl+1                                #pocet uzlu
7     elements=np.zeros((nEl,2),dtype=np.int32)  #uzly elementu
8     coordinates=np.zeros((nNod,2),dtype=np.float64) #souradnice uzlu
9     h=L/nEl                                    #delka elementu
10    for i in range(nEl):
11        elements[i,:]=np.array([i,i+1])
12    for i in range(nEl+1):
13        coordinates[i,:]=np.array([i*h,0])
14    return elements,coordinates
15
16 #startovaci procedura
17 def getStart(M,K,C,deltaT,u0,v0,F0):
18     P_eff = F0-np.dot(C,v0)-np.dot(K,u0)      #prava strana
19     a0 = np.linalg.solve(M,P_eff)             #pocatecni zrychleni
20     uStart = u0-deltaT*v0+deltaT**2/2*a0      #vektor pocatecnich posuvu
21     return uStart
22
23 #explicitni algoritmus
24 def runExplicit(nEl,n,T,deltaT,K,C,M,F,u0,u_t0,u_t1):
25     u_num = np.zeros((nEl+1,n+1),dtype=np.float64) #vektor posuvu
26     u_num[:,0] += u0.transpose()                 #aplikace PP do vekt. pos.
27     T_num = np.zeros(n+1,dtype=np.float64)       #t vyp. v jednotl. krocich
28     T_total= deltaT                               #celkovy t vypoctu
29     i = 1
30     while (T_total < T) or (T_total == T):
31         print('t = %.4f s' % T_total)
32         X = F-np.dot(K,u_t1)                     #substituce
33         Y = 2*u_t1-u_t0                          #substituce
34         Z = np.dot(C,u_t0)                       #substituce
35         M_eff =M + deltaT/2*C                    #substituce
36         F_eff = deltaT**2*X+np.dot(M,Y)+deltaT/2*Z #substituce
```

```

37     u_t2 = np.linalg.solve(M_eff,F_eff)      #reseni soust. "A*x = b"
38     u_num[:,i] += u_t2.transpose()           #ulozeni vysledku
39     T_num[i] += T_total                      #ulozeni vysledku
40     T_total += deltaT                       #ulozeni vysledku
41     if T_total == 0.05 or T_total > 0.05:    #skokove odebrani zatizeni
42         F=np.zeros(nEl+1,dtype=np.float64)
43         u_t0 = u_t1
44         u_t1 = u_t2
45         i += 1
46     return u_num,T_num
47
48 #material, geometrie
49 L = 1.                                     #delka struny [m]
50 d = 1e-3                                  #prumer struny [m]
51 rho = 7850.                               #hustota [kg*m^-3]
52 E = 2.1e11                                #modul pruznosti[Pa]
53 alpha = 80.                               #koeficient vnejsiho tlumeni
54 beta = 0.                                 #koeficient vnitriho tlumeni
55 P = 4775.                                 #predepinaci sila [N]
56 Q = 100.                                  #zatizeni
57 T_vyp = 0.1                               #delka analyzy[s]
58 eta = 0.8                                #zmenseni casoveho kroku
59 #sit
60 nEl = 80                                  #pocet elementu
61 h = L/nEl                                #delka elementu
62 A = np.pi*d**2/4                         #plocha prurezu elementu
63 mh = rho*A*h                              #hmotnost elementu
64 omega = 2*np.sqrt(3*P/(mh*h))             #max. vlastni kruh. frekv. prvku
65 elements,coordinates = getMesh(L,nEl)     #elementy,souradnice uzlu
66 #casovy krok
67 period = 2*np.pi/omega                   #perioda
68 T = T_vyp                                 #cas vypoctu
69 deltaT = eta*period/np.pi                 #casovy krok
70 n = int(T/deltaT)                         #pocet vypocetnich kroku
71 #lokalni matice
72 M_loc = (mh/2)*np.array([[1,0],[0,1]],dtype=np.float64) #lok. mat. hmotnosti
73 K_loc = (P/h)*np.array([[1,-1],[-1,1]],dtype=np.float64) #lok. mat. tuhosti
74 #sestaveni globalnich matic
75 nE = elements.shape[0]                   #pocet elementu
76 nCoord = coordinates.shape[0]            #pocet uzlu
77 nDOFs = nCoord                           #pocet st.volnosti
78 K = np.zeros((nDOFs,nDOFs),dtype=np.float64)
79 M = np.zeros((nDOFs,nDOFs),dtype=np.float64)
80 for i in range(nE):
81     indEl=elements[i,:]                  #uzly jednotl. elementu
82     iDOF=indEl                           #vektor posuvu uzlu 1 elementu
83     K[np.ix_(iDOF,iDOF)]+=K_loc          #glob. mat. tuhosti
84     M[np.ix_(iDOF,iDOF)]+=M_loc          #glob. mat. hmotnosti
85 C = alpha*M + beta*K                    #matice tlumeni
86 #vektor zatizeni
87 F=np.zeros(nDOFs,dtype=np.float64)
88 F[:] = -Q/nCoord

```



```

89 F[0]=0
90 F[-1]=0
91 #okrajove podminky
92 K[0,:] = 0.
93 K[:,0] = 0.
94 K[0,0] = 1.
95 K[-1,:] = 0.
96 K[:,-1] = 0.
97 K[-1,-1] = 1.
98 M[0,:] = 0.
99 M[:,0] = 0.
100 M[0,0] = 1.
101 M[-1,:] = 0.
102 M[:,-1] = 0.
103 M[-1,-1] = 1.
104 #pocatecni podminky
105 u0 = np.zeros(nCoord,dtype=np.float64)
106 v0 = np.zeros(nCoord,dtype=np.float64)
107 F0 = np.zeros(nCoord,dtype=np.float64)
108 F0[:] = Q/nCoord
109 F0[0] = 0
110 F0[-1] = 0
111 #znaceni:
112 #u_t0 = u_t-deltaT
113 #u_t1 = u_t
114 #u_t2 = u_t+deltaT
115 #startovaci procedura
116 uStart = getStart(M,K,C,deltaT,u0,v0,F0)
117 u_t1 = u0
118 u_t0 = uStart
119 #explicitni algoritmus
120 u_num,T_num = runExplicit(nEl,n,T,deltaT,K,C,M,F,u0,u_t0,u_t1)

```

---



## Příloha C

# Prut-pružina

---

```
1 import numpy as np
2 import Start as st
3 #nacteni site z textoveho souboru
4 def readMesh():
5     elements = np.loadtxt('elem.txt',dtype=np.int)
6     coordinates = np.loadtxt('coord.txt')
7     return elements,coordinates
8 #transformacni matice
9 def T(sin,cos):
10     return np.array([[cos,sin,0,0],\
11                     [-sin,cos,0,0],\
12                     [0,0,cos,sin],\
13                     [0,0,-sin,cos]],dtype=np.float64)
14 #matice hmotnosti
15 def M_e(elem_mass):
16     return (elem_mass/3)*np.array([[1,0,0,0],\
17                                   [0,1,0,0],\
18                                   [0,0,1,0],\
19                                   [0,0,0,1]],dtype=np.float64)
20 #matice tuhosti
21 def K_L(Young_mod,area,length):
22     return (Young_mod*area/length)*np.array([[1,0,-1,0],\
23                                             [0,0,0,0],\
24                                             [-1,0,1,0],\
25                                             [0,0,0,0]],dtype=np.float64)
26 #matice geometricke tuhosti
27 def K_NL(force,length):
28     return (force/length)*np.array([[1,0,-1,0],\
29                                   [0,1,0,-1],\
30                                   [-1,0,1,0],\
31                                   [0,-1,0,1]],dtype=np.float64)
32 #transformace matice
33 def M_Transform(matrix):
34     return np.dot(np.dot(T(sin_phi,cos_phi).transpose(),matrix),T(sin_phi,cos_phi))
35 #transformace vektoru
36 def V_Transform(vector):
```

```

37     return np.dot(T(sin_phi,cos_phi),vector)
38 #=====
39 #zadani
40 #=====
41 A = 1.
42 rho = 0.1                #A*rho [lb*sec^2/in^2]
43 E = 1e7                  #A*E [lb]
44 alpha = 0.               #souc. vnejsiho tlumeni
45 beta = 0.               #souc. vnitriho tlumeni
46 T_vyp = 8.              #cas vypoctu [s]
47 delta = 0.02            #faktor pro velikost dt
48 k_pruzina = 6.          #tuhost pruziny [lb/in]
49 Fc = 5.                 #zatezujici sila [lb]
50 #=====
51 # nacteni site
52 #=====
53 elements,coordinates = rd.readMesh()      #elementy,uzly
54 nE = 1                                    #pocet elementu
55 nCoord = coordinates.shape[0]             #pocet uzlu
56 nDOFs=nCoord*2                           #pocet st.volnosti
57 #delka,prumer prvku
58 h0 = np.zeros(nE,dtype=np.float64)
59 for k in range(nE):
60     h_XY = coordinates[k+1,:]-coordinates[k,:] #[x2-x1,y2-y1]=[hx,hy]
61     h = np.sqrt(np.sum(h_XY**2))               #delka elementu ze souradnic
62     h0[k] += h                                #vektor pocatecnich delek el.
63 #=====
64 # #casovy krok
65 #=====
66 mh = rho*A*min(h0)                        #hmotnost elementu [kg]
67 omega = 2*np.sqrt(A*E/(mh*min(h0)))        #vlastni kruhova frekvence prvku
68 period = 2*np.pi/omega                   #perioda
69 deltaT = delta*period/np.pi               #casovy krok
70 n = int(T_vyp/deltaT)                     #pocet vypocetnich kroku
71 #=====
72 # #pocatecni podminky
73 #=====
74 u0 = np.zeros(nDOFs,dtype=np.float64)     #pocatecni vychylka
75 v0 = np.zeros(nDOFs,dtype=np.float64)     #pocatecni rychlost
76 F0 = np.zeros(nDOFs,dtype=np.float64)     #pocatecni zatizeni
77 #=====
78 # zatizeni
79 #=====
80 F=np.zeros(nDOFs,dtype=np.float64)        #vektor vnejsich sil
81 #=====
82 # aplikace PP do K a M - "1. vypocetni krok"
83 #=====
84 T_total = deltaT                          #celkovy cas vypoctu
85 u_num = np.zeros((nDOFs,1),dtype=np.float64) #vektor posuvu
86 u_num[:,0] += u0                          #aplikace PP do vektoru posuvu
87 T_num = np.zeros(1,dtype=np.float64)      #cas vypoctu v jednotl. krocich(pro
      vykresleni grafu)

```

```

88 M=np.zeros((nDOFs,nDOFs),dtype=np.float64)           #globalni mat. hmotnosti
89 K=np.zeros((nDOFs,nDOFs),dtype=np.float64)           #globalni mat. tuhosti
90 R = np.zeros(nDOFs,dtype=np.float64)                 #vektor vnitřních uzlových sil
91 for k in range(nE):
92     h_X = (u0[k*2+2]+coordinates[k+1,0])-(u0[k*2]+coordinates[k,0]) # x2-x1 =
93         hx...vyska elementu v ose x
94     h_Y = (u0[k*2+3]+coordinates[k+1,1])-(u0[k*2+1]+coordinates[k,1]) # y2-y1 =
95         hy...vyska elementu v ose y
96     p = np.array([coordinates[k+1,0]-coordinates[k,0],0])           #vektor
97         predstavujici puvodni element l
98     q = np.array([h_X, h_Y])                                       #vektor -
99         natoceny deformovany element l'
100     cos_phi = np.dot(p.transpose(),q)/(np.linalg.norm(p)*np.linalg.norm(q)) #cos
101         uhlu ze skalarniho soucinu
102     sin_phi = np.linalg.norm(np.cross(p,q))/(np.linalg.norm(p)*np.linalg.norm(q))
103         #sin uhlu z vektoroveho soucinu
104     if (h_Y < 0):
105         sin_phi = -1*sin_phi
106     phi = np.arccos(cos_phi)           #informativni vypocet uhlu natoceni
107     u_loc = V_Transform(u0[k*2:k*2+4]) #vektor posuvu v lok. SS
108     deltaH = u_loc[2]-u_loc[0]          #zmena delky elementu
109     h1 = h0[k]+deltaH                  #nova delka elementu
110     epsilon = (h1**2-h0[k]**2)/(2*h0[k]**2) #pomerna deformace elementu
111     N = E*A*epsilon/2                  #normalova sila v elementu od deformace
112     F_T = N*np.array([[-cos_phi],\
113         [-sin_phi],\
114         [cos_phi],\
115         [sin_phi]],dtype=np.float64) #vnitrni uzlove reakce
116         #vnitrni uzlove reakce
117     R[0+k*2]+=F_T[0]
118     R[1+k*2]+=F_T[1]
119     R[2+k*2]+=F_T[2]
120     R[3+k*2]+=F_T[3]
121     R[:2]=0
122     K_l = M_Transform(K_L(E/2,A,h)) + K_NL(N,h) #glob. matice tuhosti elementu
123     iDOF = np.array([0,1,2,3])+2*k
124     K[np.ix_(iDOF,iDOF)] += K_l           #glob. mat. tuhosti
125     M[np.ix_(iDOF,iDOF)] += M_e(mh)       #glob. mat. hmotnosti
126 #=====
127 # okrajove podminky
128 #=====
129 #vazba - pevny kloub
130 node0x = elements[0]
131 node0y = elements[0]+1
132 node0 = np.array([node0x,node0y])
133 K[node0,:]=0.0
134 K[:,node0]=0.0
135 nI=node0.shape[0]
136 K[np.ix_(node0,node0)]+=np.eye(nI,dtype=np.float64)
137 K[-1,-1] += k_pruzina
138 M[node0,:]=0.0
139 M[:,node0]=0.0
140 M[:,2] = 0.

```

```

134 M[2,:] = 0.
135 M[np.ix_(node0,node0)]+=np.eye(nI,dtype=np.float64)
136 M[2,2] = 1.
137 N_int = np.zeros((nDOFs,1),dtype=np.float64)      #vektor vnitřních sil
138 N_int[:,0] += np.dot(K,u0)
139 C = alpha*M + beta*K                               #mat. tlumení
140 #=====
141 # startovací procedura
142 #=====
143 uStart = st.getStart(R,M,K,C,deltaT,u0,v0,F0)
144 u_t1 = u0                                           #u_t1 = vektor posuvu u v case t
145 u_t0 = uStart                                       #u_t0 = u v case t-deltaT
146 #=====
147 # vypocetni algoritmus
148 #=====
149 maxSteps = 1000000
150 for i in range(maxSteps):
151     X = F-np.dot(K,u_t1)-R                         #subst.
152     Y = 2*u_t1-u_t0                                #subst.
153     Z = np.dot(C,u_t0)                             #subst.
154     M_eff = M + deltaT/2*C                          #leva strana rovnice pro vypocet
155     u_{t+deltaT}
156     F_eff = deltaT**2*X+np.dot(M,Y)+deltaT/2*Z      #prava strana rovnice
157     u_t2 = np.linalg.solve(M_eff,F_eff)            #reseni mat. rce M_eff*u_{t+deltaT}
158     = F_eff ("A*x = b")
159     #aplikace zatizeni
160     if T_total<0.1:
161         F[-1]+=-Fc*deltaT/0.1
162     u_t2[-2] = 0.
163     u_t2[0] = 0.
164     u_t2[1] = 0.
165     u = np.zeros((nDOFs,1),dtype=np.float64)      #pomocny vektor
166     u[:,0] += u_t2
167     u_inc = u_t2-u_num[:,i]
168     u_num = np.append(u_num,u,axis=1)
169     T_num = np.append(T_num,T_total)
170     T_total += deltaT
171     if (T_total == T_vyp) or (T_total > T_vyp):
172         break
173     else:
174         #sestaveni novych matic
175         R = np.zeros(nDOFs,dtype=np.float64)
176         K=np.zeros((nDOFs,nDOFs),dtype=np.float64)
177         for k in range(nE):
178             h_X = (u_t2[k*2+2]+coordinates[k+1,0])-(u_t2[k*2]+coordinates[k,0])
179             h_Y =
180                 (u_t2[k*2+3]/2+coordinates[k+1,1])-(u_t2[k*2+1]/2+coordinates[k,1])
181             p = np.array([coordinates[k+1,0]-coordinates[k,0],0])    #vektor
182                 predstavujici puvodni element l
183             q = np.array([h_X, h_Y])    #vektor - natoceny deformovany element l'
184             cos_phi = np.dot(p.transpose(),q)/(np.linalg.norm(p)*np.linalg.norm(q))
185             sin_phi =
186                 np.linalg.norm(np.cross(p,q))/(np.linalg.norm(p)*np.linalg.norm(q))

```

```

181         if (h_Y < 0):
182             sin_phi = -1*sin_phi
183             phi = np.arccos(cos_phi)
184         if (h_Y < 0):
185             phi = -1*phi
186         u_loc = V_Transform(u_t2[k*2:k*2+4])
187         deltaH = u_loc[2]-u_loc[0]
188         h1 = h0[k]+deltaH
189         epsilon = (h1**2-h0[k]**2)/(2*h0[k]**2)
190         N = E*A*epsilon/2
191         F_T = N*np.array([[ -cos_phi], \
192                           [-sin_phi], \
193                           [cos_phi], \
194                           [sin_phi]], dtype=np.float64)
195         R[0+k*2] += F_T[0]
196         R[1+k*2] += F_T[1]
197         R[2+k*2] += F_T[2]
198         R[3+k*2] += F_T[3]
199         R[:2] = 0
200         K_L = M_Transform(K_L(E/2,A,h)) + K_NL(N,h)
201         iDOF = np.array([0,1,2,3])+2*k
202         K[np.ix_(iDOF,iDOF)] += K_L
203         print('t = %.4f s ----- fi = %.3f deg' % (T_total, phi*180/np.pi))
204     #vazba - pevny kloub
205     node0x = elements[0]
206     node0y = elements[0]+1
207     node0 = np.array([node0x,node0y])
208     K[node0,:]=0.0
209     K[:,node0]=0.0
210     K[:,2] = 0.
211     K[2,:] = 0.
212     K[-1,-1] += k_pruzina
213     nI=node0.shape[0]
214     K[np.ix_(node0,node0)] += np.eye(nI, dtype=np.float64)
215     K[2,2] = 1.
216     M[node0,:]=0.0
217     M[:,node0]=0.0
218     M[np.ix_(node0,node0)] += np.eye(nI, dtype=np.float64)
219     M[:,2] = 0.
220     M[2,:] = 0.
221     M[2,2] = 1.
222     N_i = np.dot(K,u) #vypocet reakci
223     N_int = np.append(N_int,N_i,axis=1)
224     C = alpha*M + beta*K
225     u_t0 = u_t1
226     u_t1 = u_t2 #konec smycky

```

---





## Příloha D

# Kyvadlo

### D.1 Numerické řešení analytických diferenciálních rovnic

---

```
1 from numpy import sin, cos
2 import numpy as np
3 import scipy.integrate as integrate
4
5 g = 9.81 #tíhové zrychlení [m/s^2]
6 L1 = 0.15 #delka zavesu 1 [m]
7 L2 = 0.15 #delka zavesu 2 [m]
8 M1 = 1.0 #hmotnost 1 [kg]
9 M2 = 1.0 #hmotnost 2 [kg]
10 #stavový prostor
11 def derivs(state, t):
12     dydx = np.zeros_like(state)
13     dydx[0] = state[1]
14
15     del_ = state[2] - state[0]
16     den1 = (M1 + M2)*L1 - M2*L1*cos(del_)*cos(del_)
17     dydx[1] = (M2*L1*state[1]*state[1]*sin(del_)*cos(del_) +
18               M2*g*sin(state[2])*cos(del_) +
19               M2*L2*state[3]*state[3]*sin(del_) -
20               (M1 + M2)*g*sin(state[0]))/den1
21     dydx[2] = state[3]
22     den2 = (L2/L1)*den1
23     dydx[3] = (-M2*L2*state[3]*state[3]*sin(del_)*cos(del_) +
24               (M1 + M2)*g*sin(state[0])*cos(del_) -
25               (M1 + M2)*L1*state[1]*state[1]*sin(del_) -
26               (M1 + M2)*g*sin(state[2]))/den2
27     return dydx
28 # časový krok
29 dt = 0.000005
30 t = np.arange(0.0, 5, dt)
31 #pocateční uhly natocení zavesu a pocateční uhlove rychlosti
32 th10 = 0.
33 w10 = 0.0
```

```

34 th20 = 0.0
35 w20 = 0.
36 w20 = 4.187*180/np.pi
37 # pocatecni podminky
38 state = np.radians([th10, w10, th20, w20])
39 # reseni diferencialnich rovnic
40 y = integrate.odeint(derivs, state, t)
41 #vychylky
42 x1 = L1*sin(y[:, 0])
43 y1 = -L1*cos(y[:, 0])
44 x2 = L2*sin(y[:, 2]) + x1
45 y2 = -L2*cos(y[:, 2]) + y1
46 #rychlosti
47 vx1 = L1*sin(y[:, 1])
48 vy1 = -L1*cos(y[:, 1])
49 vx2 = L2*sin(y[:, 3]) + vx1
50 vy2 = -L2*cos(y[:, 3]) + vy1

```

---

## D.2 Numerické řešení explicitní metodou pomocí MKP

---

```

1 import numpy as np
2 import Start as st
3 import reader as rd
4 #transformacni matice
5 def T2(sin,cos):
6     return np.array([[cos,sin,0,0],\
7                     [-sin,cos,0,0],\
8                     [0,0,cos,sin],\
9                     [0,0,-sin,cos]],dtype=np.float64)
10 #matice hmotnosti elementu
11 def M_e(elem_mass):
12     return (elem_mass/2)*np.array([[1,0,0,0],\
13                                   [0,1,0,0],\
14                                   [0,0,1,0],\
15                                   [0,0,0,1]],dtype=np.float64)
16 #matice elasticke tuhosti elementu
17 def K_L(Young_mod,area,length):
18     return (Young_mod*area/length)*np.array([[1,0,-1,0],\
19                                             [0,0,0,0],\
20                                             [-1,0,1,0],\
21                                             [0,0,0,0]],dtype=np.float64)
22 #matice geometricke tuhosti elementu
23 def K_NL(force,length):
24     return (force/length)*np.array([[1,0,-1,0],\
25                                   [0,1,0,-1],\
26                                   [-1,0,1,0],\
27                                   [0,-1,0,1]],dtype=np.float64)
28 #transformace matice
29 def M_Transform(matrix):

```



```

80 #=====
81 # zatizeni
82 #=====
83 F=np.zeros(nDOFs,dtype=np.float64)          #stejne jako pocatecni zatizeni
84 for i in range(nCoord):
85     F[2*i] = (mh/2+m_pend)*g
86 F[0]=0.
87 F[2] += (mh/2)*g
88 #=====
89 # aplikace PP do K a M - "1. vypocetni krok"
90 #=====
91 T_total = deltaT                               #celkovy cas vypoctu
92 u_num = np.zeros((nDOFs,1),dtype=np.float64)  #vektor posuvu
93 u_num[:,0] += u0                               #aplikace PP do vektoru posuvu
94 T_num = np.zeros(1,dtype=np.float64)          #cas vypoctu v jednotl.
95     krocich(pro vykresleni grafu)
96 K=np.zeros((nDOFs,nDOFs),dtype=np.float64)    #globalni mat. tuhosti
97 M=np.zeros((nDOFs,nDOFs),dtype=np.float64)    #globalni mat. hmotnosti
98 R = np.zeros(nDOFs,dtype=np.float64)          #vektor vnitrnich uzlovych sil
99 for k in range(nE):
100     h_X = (u0[k*2+2]+coordinates[k+1,0])-(u0[k*2]+coordinates[k,0]) # x2-x1 =
101         hx...vyska elementu v ose x
102     h_Y = (u0[k*2+3]+coordinates[k+1,1])-(u0[k*2+1]+coordinates[k,1]) # y2-y1 =
103         hy...vyska elementu v ose y
104     p = np.array([coordinates[k+1,0]-coordinates[k,0],0]) #vektor predstavujici
105         puvodni element l
106     q = np.array([h_X, h_Y]) #vektor - natoceny
107         deformovany element l'
108     cos_phi = np.dot(p.transpose(),q)/(np.linalg.norm(p)*np.linalg.norm(q)) #cos
109         uhlu ze skalarniho soucinu
110     sin_phi = np.linalg.norm(np.cross(p,q))/(np.linalg.norm(p)*np.linalg.norm(q))
111         #sin uhlu z vektoroveho soucinu
112     if (h_Y < 0):
113         sin_phi = -1*sin_phi
114     u_loc = V_Transform(u0[k*2:k*2+4]) #vektor posuvu v lok. SS
115     deltaH = u_loc[2]-u_loc[0] #zmena delky elementu
116     h1 = h0[k]+deltaH #nova delka elementu
117     epsilon = (h1**2-h0[k]**2)/(2*h0[k]**2) #pomerna deformace
118     N = E*A*epsilon/2 #normalova sila v elementu od deformace
119     F_T = N*np.array([[ -cos_phi],\
120         [-sin_phi],\
121         [cos_phi],\
122         [sin_phi]],dtype=np.float64) #vnitrni uzlove reakce
123     R[0+k*2]+=F_T[0] #vnitrni uzlove reakce
124     R[1+k*2]+=F_T[1]
125     R[2+k*2]+=F_T[2]
126     R[3+k*2]+=F_T[3]
127     R[:2]=0
128     K_l = M_Transform(K_L(E,A,h)) + K_NL(N,h) #glob. matice tuhosti elementu
129     iDOF = np.array([0,1,2,3])+2*k
130     K[np.ix_(iDOF,iDOF)]+=K_l #glob. mat. tuhosti

```

```

125     M[np.ix_(iDOF,iDOF)]+=M_e(mh)                #glob. mat. hmotnosti
126     #=====
127     # okrajove podminky
128     #=====
129     node0x = elements[0,0]
130     node0y = elements[0,0]+1
131     node0 = np.array([node0x,node0y])
132     K[node0,:]=0.0
133     K[:,node0]=0.0
134     nI=node0.shape[0]
135     K[np.ix_(node0,node0)]+=np.eye(nI,dtype=np.float64)
136     M += m_pend*np.eye(nDOFs,dtype=np.float64)    #aplikace hmotnosti kyvadla
137     M[node0,:]=0.0
138     M[:,node0]=0.0
139     M[np.ix_(node0,node0)]+=np.eye(nI,dtype=np.float64)
140
141     C = alpha*M + beta*K                          #mat. tlumeni
142     #=====
143     # startovaci procedura
144     #=====
145     uStart,a0 = st.getStart(R,M,K,C,deltaT,u0,v0,F0)
146     u_t1 = u0                                     #u_t1 = vektor posuvu u v case t
147     u_t0 = uStart                                #u_t0 = u v case t-deltaT
148     #=====
149     # vypocetni algoritmus
150     #=====
151     maxIter = 1000000
152     for i in range(maxIter):
153         X = F-np.dot(K,u_t1)-R                    #subst.
154         Y = 2*u_t1-u_t0                           #subst.
155         Z = np.dot(C,u_t0)                         #subst.
156         M_eff = M + deltaT/2*C                     #leva strana rovnice pro vypocet
157         F_eff = deltaT**2*X+np.dot(M,Y)+deltaT/2*Z #prava strana rovnice
158         u_t2 = np.linalg.solve(M_eff,F_eff)        #reseni mat. rce M_eff*u_{t+deltaT}
159         u = np.zeros((nDOFs,1),dtype=np.float64)  #pomocny vektor
160         u[:,0] += u_t2
161         u_num = np.append(u_num,u,axis=1)
162         T_num = np.append(T_num,T_total)
163         T_total += deltaT
164         if (T_total == T_vyp) or (T_total > T_vyp):
165             break
166         else:                                       #sestaveni novych matic
167             R = np.zeros(nDOFs,dtype=np.float64)
168             K=np.zeros((nDOFs,nDOFs),dtype=np.float64)
169             for k in range(nE):
170                 h_X = (u_t2[k*2+2]/2+coordinates[k+1,0])-(u_t2[k*2]/2+coordinates[k,0])
171                 h_Y =
172                     (u_t2[k*2+3]/2+coordinates[k+1,1])-(u_t2[k*2+1]/2+coordinates[k,1])
173                 p = np.array([coordinates[k+1,0]-coordinates[k,0],0]) #vektor
174                     predstavujici puvodni element l

```

```

173     q = np.array([h_X, h_Y])                                #vektor - natoceny
174     deformovany element l'
175     cos_phi = np.dot(p.transpose(),q)/(np.linalg.norm(p)*np.linalg.norm(q))
176     sin_phi =
177         np.linalg.norm(np.cross(p,q))/(np.linalg.norm(p)*np.linalg.norm(q))
178     if (h_Y < 0):
179         sin_phi = -1*sin_phi
180     phi = np.arccos(cos_phi)
181     if (h_Y < 0):
182         phi = -1*phi
183     u_loc = V_Transform(u_t2[k*2:k*2+4])
184     deltaH = u_loc[2]-u_loc[0]
185     h1 = h0[k]+deltaH
186     epsilon = (h1**2-h0[k]**2)/(2*h0[k]**2)
187     N = E*A*epsilon/2
188     F_T = N*np.array([[-cos_phi],\
189                     [-sin_phi],\
190                     [cos_phi],\
191                     [sin_phi]],dtype=np.float64)
192     R[0+k*2]+=F_T[0]
193     R[1+k*2]+=F_T[1]
194     R[2+k*2]+=F_T[2]
195     R[3+k*2]+=F_T[3]
196     R[:2]=0
197     K_l = M_Transform(K_L(E/2,A,h)) + K_NL(N,h)
198     iDOF = np.array([0,1,2,3])+2*k
199     K[np.ix_(iDOF,iDOF)]+=K_l
200     print('t = %.4f s ----- fi = %.3f deg' % (T_total,phi*180/np.pi))
201     node0x = elements[0,0]
202     node0y = elements[0,0]+1
203     node0 = np.array([node0x,node0y])
204     K[node0,:]=0.0
205     K[:,node0]=0.0
206     nI=node0.shape[0]
207     K[np.ix_(node0,node0)]+=np.eye(nI,dtype=np.float64)
208     C = alpha*M + beta*K
209     u_t0 = u_t1
210     u_t1 = u_t2                                #konec smycky
211     #=====
212     # rychlost
213     #=====
214     v = np.zeros_like(u_num,dtype=np.float64)
215     for i in range(nDOFs):
216         for j in range(n):
217             if (j == 0):
218                 v[i,j] += v0[i]
219             elif (j == n-1):
220                 v[i,j] += (u_num[i,j]-u_num[i,j-1])/(deltaT)
221             else:
222                 v[i,j] += (u_num[i,j+1]-u_num[i,j-1])/(2*deltaT)
223     #=====
224     # zrychleni

```

```
223 #=====
224 a = np.zeros_like(u_num,dtype=np.float64)
225 for i in range(nDOfs):
226     for j in range(n):
227         if (j == 0):
228             a[i,j] += a0[i]
229         elif (j == n-1):
230             a[i,j] += a[i,j-1]
231         else:
232             a[i,j] += (u_num[i,j-1]-2*u_num[i,j]+u_num[i,j+1])/(deltaT**2)
```

---





## Příloha E

# Bezpečnostní pás 2D

---

```
1 import numpy as np
2 import Start as st
3 #nacteni site z exportovaneho textoveho souboru z LS-DYNY
4 def readMesh():
5     elements =
6         np.loadtxt('elements.txt',dtype=np.int,comments='*',skiprows=5,usecols=(2,3))
7     coordinates = np.loadtxt('coordinates_LS.txt')
8     return elements,coordinates
9 #transformacni matice
10 def T2(sin,cos):
11     return np.array([[cos,sin,0,0],\
12                     [-sin,cos,0,0],\
13                     [0,0,cos,sin],\
14                     [0,0,-sin,cos]],dtype=np.float64)
15 #matice hmotnosti elementu
16 def M_e(elem_mass):
17     return (elem_mass/2)*np.array([[1,0,0,0],\
18                                   [0,1,0,0],\
19                                   [0,0,1,0],\
20                                   [0,0,0,1]],dtype=np.float64)
21 #matice elasticke tuhosti elementu
22 def K_L(Young_mod,area,length):
23     return (Young_mod*area/length)*np.array([[1,0,-1,0],\
24                                               [0,0,0,0],\
25                                               [-1,0,1,0],\
26                                               [0,0,0,0]],dtype=np.float64)
27 #matice geometricke tuhosti elementu
28 def K_NL(force,length):
29     return (force/length)*np.array([[1,0,-1,0],\
30                                    [0,1,0,-1],\
31                                    [-1,0,1,0],\
32                                    [0,-1,0,1]],dtype=np.float64)
33 #transformace matice
34 def M_Transform(matrix):
35     return
36         np.dot(np.dot(T2(sin_phi,cos_phi).transpose(),matrix),T2(sin_phi,cos_phi))
```

```

35 #transformace vektoru
36 def V_Transform(vector):
37     return np.dot(T2(sin_phi,cos_phi),vector)
38 #=====
39 # #material, geometrie
40 #=====
41 d = 8.6e-3                #prumer [m]
42 rho = 1100.               #hustota [kg*m^-3]
43 g = 9.81                  #m/s^2
44 E = 1.05e9                #[Pa]
45 mu = 0.39                 #Poisson. c.
46 alpha = 0.                #sourc. tlumeni
47 beta = 0.000003           #sourc. tlumeni
48 A = np.pi*d**2/4.         #plocha prurezu elementu [m^2]
49 T_vyp = 0.18              #cas vypoctu [s]
50 delta = 0.8               #faktor pro velikost dt
51 natoceni = 10.            #uhel natoceni sedadla ve stupnich
52 #=====
53 # nacteni site
54 #=====
55 elements,coordinates = rd.readMesh()      #elementy,uzly
56 coordinates = coordinates/1000
57 elements = elements-1
58 nE = elements.shape[0]                   #pocet elementu
59 nCoord = coordinates.shape[0]            #pocet uzlu
60 nDOFs=nCoord*2                           #pocet st.volnosti
61 #delka,prumer prvku
62 h0 = np.zeros(nE,dtype=np.float64)
63 for k in range(nE):
64     h_XY = coordinates[k+1,:]-coordinates[k,:] #[x2-x1,y2-y1]=[hx,hy]
65     h = np.sqrt(np.sum(h_XY**2))                #delka elementu ze
66     souradnic=sqrt(hx**2+hy**2)
67     h0[k] += h
68 #=====
69 # #casovy krok
70 #=====
71 mh = rho*A*min(h0)                        #hmotnost elementu [kg]
72 omega = 2*np.sqrt(A*E/(mh*min(h0)))        #vlastni kruhova frekvence prvku
73 period = 2*np.pi/omega                   #perioda
74 deltaT = delta*period/np.pi               #casovy krok
75 n = int(T_vyp/deltaT)                      #pocet vypocetnich kroku
76 #=====
77 # #pocatecni podminky
78 #=====
79 u0 = np.zeros(nDOFs,dtype=np.float64)     #pocatecni vychylka
80 v0 = np.zeros(nDOFs,dtype=np.float64)     #pocatecni rychlost
81 F0 = np.zeros(nDOFs,dtype=np.float64)     #pocatecni zatizeni
82 #=====
83 # zatizeni
84 #=====
85 F=np.zeros(nDOFs,dtype=np.float64)        #stejne jako pocatecni zatizeni

```

```

86 # aplikace PP do K a M - "1. vypocetni krok"
87 #=====
88 T_total = deltaT #celkovy cas vypoctu
89 u_num = np.zeros((nDOFs,1),dtype=np.float64) #vektor posuvu
90 u_num[:,0] += u0 #aplikace PP do vektoru posuvu
91 T_num = np.zeros(1,dtype=np.float64) #cas vypoctu v jednotl.
    krocich(pro vykresleni grafu)
92
93 K=np.zeros((nDOFs,nDOFs),dtype=np.float64) #globalni mat. tuhosti
94 M=np.zeros((nDOFs,nDOFs),dtype=np.float64) #globalni mat. hmotnosti
95 R = np.zeros(nDOFs,dtype=np.float64) #vektor vnitrnich uzlovych sil
96
97 for k in range(nE):
98     h_X = (u0[k*2+2]+coordinates[k+1,0])-(u0[k*2]+coordinates[k,0]) # x2-x1 =
        hx...vyska elementu v ose x
99     h_Y = (u0[k*2+3]+coordinates[k+1,1])-(u0[k*2+1]+coordinates[k,1]) # y2-y1 =
        hy...vyska elementu v ose y
100     p = np.array([coordinates[-1,0]-coordinates[0,0],0]) #vektor predstavujici
        puvodni element l
101     q = np.array([h_X, h_Y]) #vektor - natoceny
        deformovany element l'
102     cos_phi = np.dot(p.transpose(),q)/(np.linalg.norm(p)*np.linalg.norm(q)) #cos
        uhlu ze skalarniho soucinu
103     sin_phi = np.linalg.norm(np.cross(p,q))/(np.linalg.norm(p)*np.linalg.norm(q))
        #sin uhlu z vektoroveho soucinu
104     if (h_Y < 0):
105         sin_phi = -1*sin_phi
106     u_loc = V_Transform(u0[k*2:k*2+4]) #vektor posuvu v lok. SS
107     deltaH = u_loc[2]-u_loc[0] #zmena delky elementu
108     h1 = h0[k]+deltaH #nova delka elementu
109     N = deltaH*E*A/h0[k] #normalova sila v elementu od deformace
110     F_T = N*np.array([[ -cos_phi],\
        [-sin_phi],\
        [cos_phi],\
        [sin_phi]],dtype=np.float64) #vnitrni uzlove reakce
        #vnitrni uzlove reakce
111     R[0+k*2]+=F_T[0]
112     R[1+k*2]+=F_T[1]
113     R[2+k*2]+=F_T[2]
114     R[3+k*2]+=F_T[3]
115     R[:2]=0
116     R[-2:]=0
117
118
119
120
121     K_l = M_Transform(K_L(E,A,h0[k])) + K_NL(N,h0[k]) #glob. matice tuhosti
        elementu
122     iDOF = np.array([0,1,2,3])+2*k
123     K[np.ix_(iDOF,iDOF)] += K_l #glob. mat. tuhosti
124     M[np.ix_(iDOF,iDOF)] += M_e(mh) #glob. mat. hmotnosti
125
126 for i in range(nE):
127     if i < nE/2 or i == nE/2:
128         M[2*i,2*i] += 1*i*3.08
129         M[2*i+1,2*i+1] += 1*i*3.08

```

```

130     else:
131         M[2*i,2*i] += 1*(15.4-(i-nE/2)*3.08)
132         M[2*i+1,2*i+1] += 1*(15.4-(i-nE/2)*3.08)
133 C = alpha*M + beta*K #mat. tlumeni
134
135 N_int = np.zeros((nDOFs,1),dtype=np.float64) #vektor vnitřních sil
136 N_int[:,0] += np.dot(K,u0)
137 #=====
138 # startovací procedura
139 #=====
140 uStart = st.getStart(R,M,K,C,deltaT,u0,v0,F0)
141 u_t1 = u0 #u_t1 = vektor posuvu u v case t
142 u_t0 = uStart #u_t0 = u v case t-deltaT
143 #=====
144 # výpočetní algoritmus
145 #=====
146 maxIter = 10000000
147 for i in range(maxIter):
148     #aplikace predepsaneho zrychleni
149     if T_total < 0.09:
150         a = 1744*T_total
151     elif T_total == 0.09 or T_total > 0.09:
152         a = 157-1744*(T_total-0.09)
153     elif T_total == 0.18 or T_total > 0.18:
154         a = 0.
155     #aplikace zatizeni do vektoru vnejsich sil
156     for j in range(nCoord):
157         F[2*j+1] = M[2*j,2*j]*a*np.cos(np.radians(natoceni))
158         F[2*j] = -M[2*j,2*j]*a*np.sin(np.radians(natoceni))
159     F[:2] = 0.
160     F[-2:] = 0.
161     X = F-np.dot(K,u_t1)-R #subst.
162     Y = 2*u_t1-u_t0 #subst.
163     Z = np.dot(C,u_t0) #subst.
164     M_eff = M + deltaT/2*C #leva strana rovnice pro výpočet
165         u_{t+deltaT}
166     F_eff = X*deltaT**2+np.dot(M,Y)+deltaT/2*Z #prava strana rovnice
167     u_t2 = np.linalg.solve(M_eff,F_eff) #reseni mat. rce M_eff*u_{t+deltaT}
168         = F_eff ("A*x = b")
169     #okrajove podminky
170     u_t2[:2] = 0.
171     u_t2[-2:] = 0.
172     u = np.zeros((nDOFs,1),dtype=np.float64) #pomocny vektor
173     u[:,0] += u_t2
174     u_num = np.append(u_num,u,axis=1)
175     T_num = np.append(T_num,T_total)
176     T_total += deltaT
177
178     R = np.zeros(nDOFs,dtype=np.float64)
179     K=np.zeros((nDOFs,nDOFs),dtype=np.float64)
180     for k in range(nE):
181         h_X = (u_t2[k*2+2]/2+coordinates[k+1,0])-(u_t2[k*2]/2+coordinates[k,0])

```

```

180     h_Y = (u_t2[k*2+3]/2+coordinates[k+1,1])-(u_t2[k*2+1]/2+coordinates[k,1])
181     p = np.array([coordinates[-1,0]-coordinates[0,0],0])    #vektor
182     predstavujici_puvodni_element_hx
183     q = np.array([h_X, h_Y])    #vektor - natoceny deformovany element l'
184     cos_phi = np.dot(p.transpose(),q)/(np.linalg.norm(p)*np.linalg.norm(q))
185     sin_phi =
186         np.linalg.norm(np.cross(p,q))/(np.linalg.norm(p)*np.linalg.norm(q))
187     if (h_Y < 0):
188         sin_phi = -1*sin_phi
189     u_loc = V_Transform(u_t2[k*2:k*2+4])
190     deltaH = u_loc[2]-u_loc[0]
191     deltaD = -mu*d*deltaH/h0[k]
192     d1 = d+deltaD
193     A1 = np.pi*d1**2/4
194     h1 = h0[k]+deltaH
195     epsilon = (h1**2-h0[k]**2)/(2*h0[k]**2)
196     N = E*A1*epsilon/2
197     F_T = N*np.array([[ -cos_phi],\
198                     [-sin_phi],\
199                     [cos_phi],\
200                     [sin_phi]],dtype=np.float64)
201     R[0+k*2]+=F_T[0]
202     R[1+k*2]+=F_T[1]
203     R[2+k*2]+=F_T[2]
204     R[3+k*2]+=F_T[3]
205     K_l = M_Transform(K_L(E/2,A1,h1)) + K_NL(N,h1)
206     iDOF = np.array([0,1,2,3])+2*k
207     K[np.ix_(iDOF,iDOF)]+=K_l
208     print('t = %.4f s ----- a = %.4f m/s**2' % (T_total,a))
209     if (T_total == T_vyp) or (T_total > T_vyp):
210         break
211     else:
212         #sestaveni novych matic
213         C = alpha*M + beta*K
214         #vypocet reakci
215         Ri = np.zeros((nDOFs,1))
216         Ri[:,0]+=R
217         N_i = np.dot(K,u)+Ri
218         N_int = np.append(N_int,N_i,axis=1)
219
220     u_t0 = u_t1
221     u_t1 = u_t2    #konec smycky
222
223     #=====
224     # rychlost
225     #=====
226     v = np.zeros_like(u_num,dtype=np.float64)
227     for i in range(nDOFs):
228         for j in range(n+1):
229             if (j == 0):
230                 v[i,j] += v0[i]
231             elif (j == n):
232                 v[i,j] += (u_num[i,j]-u_num[i,j-1])/(deltaT)
233             else:

```

```
230         v[i,j] += (u_num[i,j+1]-u_num[i,j-1])/(2*deltaT)
231 #=====
232 # zrychleni
233 #=====
234 a0 = np.zeros_like(v0,dtype=np.float64)
235 a = np.zeros_like(u_num,dtype=np.float64)
236 for i in range(nDOFs):
237     for j in range(n+1):
238         if (j == 0):
239             a[i,j] += a0[i]
240         elif (j == n):
241             a[i,j] += a[i,j-1]
242         else:
243             a[i,j] += (u_num[i,j-1]-2*u_num[i,j]+u_num[i,j+1])/(deltaT**2)
```

---

## Příloha F

### CD nosič

- text diplomové práce
- vizualizace 3D simulace v LS-DYNA:
  - pohled z boku
  - pohled shora
  - pohled zepředu





# Seznam obrázků

1.1	Předepsaný průběh zatížení . . . . .	20
1.2	Konfigurace testu #1 . . . . .	21
1.3	Konfigurace testu #2 . . . . .	22
1.4	50 percentilová figurína Hybrid III [21] . . . . .	23
2.1	Schéma bezpečnostních pásů; zleva: 2 bodový; Y-pás; 3 bodový; 4 bodový; 5 bodový . . . . .	25
2.2	Referenční bod sedadla . . . . .	26
2.3	Linearizovaná materiálová charakteristika pásu . . . . .	27
2.4	Linearizovaná materiálová křivka nylonového pásu . . . . .	28
2.5	Rozměry bezpečnostního pásu . . . . .	28
2.6	Profil průřezu bezpečnostního pásu . . . . .	28
2.7	Běžná spona bezpečnostního pásu [22],[23] . . . . .	29
2.8	Tlačítková spona bezpečnostního pásu [24] . . . . .	29
2.9	Rotační spona bezpečnostního pásu [25],[26] . . . . .	30
3.1	Příklad časového průběhu složky vektoru posuvů $u_i$ . . . . .	31
3.2	Vliv velikosti časového kroku . . . . .	34
3.3	Vliv velikosti časového kroku - detail . . . . .	35
3.4	Schéma úlohy netlumeného kmitání s 1 stupněm volnosti . . . . .	37
3.5	Analytické a numerické řešení netlumeného kmitání . . . . .	39
3.6	Schéma úlohy tlumeného kmitání s 1 stupněm volnosti . . . . .	39
3.7	Analytické a numerické řešení tlumeného kmitání . . . . .	41
3.8	Stupně volnosti elementu v lokálním souřadném systému . . . . .	43
3.9	Schéma natočení elementu . . . . .	47
3.10	Schéma úlohy kmitání struny . . . . .	49
3.11	Průběh zatížení struny . . . . .	49
3.12	Stupně volnosti elementu v lokálním souřadném systému . . . . .	50
3.13	Časový průběh výchylky středového bodu struny . . . . .	51
3.14	Časový průběh výchylky středového bodu struny - vliv tlumení . . . . .	52

3.15	Schéma úlohy kmitání systému prut-pružina . . . . .	53
3.16	Průběh zatížení prutu . . . . .	53
3.17	Časový průběh výchylky bodu B . . . . .	55
3.18	Schéma úlohy dvojitého kyvadla . . . . .	56
3.19	Aplikace počátečních podmínek dvojitého kyvadla . . . . .	56
3.20	Stupně volnosti diskretizovaného kyvadla . . . . .	57
3.21	Časový průběh výchylky bodu 1 v ose x . . . . .	60
3.22	Časový průběh výchylky bodu 1 v ose y . . . . .	60
3.23	Časový průběh rychlosti bodu 1 v ose y . . . . .	61
3.24	Časový průběh rychlosti bodu 1 v ose x . . . . .	62
3.25	Časový průběh rychlosti bodu 1 v ose x s tlumením . . . . .	62
3.26	Časový průběh rychlosti bodu 1 v ose x s tlumením - detail . . . . .	63
3.27	Časový průběh zrychlení bodu 1 v ose x . . . . .	63
3.28	Časový průběh zrychlení bodu 1 v ose x s tlumením . . . . .	64
3.29	Časový průběh zrychlení bodu 1 v ose x s tlumením - detail . . . . .	64
3.30	Časový průběh zrychlení bodu 1 v ose y . . . . .	65
3.31	Časový průběh zrychlení bodu 1 v ose y s tlumením . . . . .	65
3.32	Časový průběh zrychlení bodu 1 v ose y s tlumením - detail . . . . .	66
4.1	3D výchozí model . . . . .	67
4.2	Rozměry modelu sedadla . . . . .	68
4.3	3D výpočetní model se zjednodušeným modelem pásu; vpravo pohled shora . . . . .	69
4.4	Geometrie pásu pro zjednodušený výpočet . . . . .	70
4.5	Skutečná diskretizovaná geometrie pásu . . . . .	70
4.6	Schéma rozdělení hmotnosti figuríny do uzlů na rozvinutém pásu . . . . .	71
4.7	Zatížení pásu . . . . .	73
4.8	Časový průběh výsledných reakcí v ukotvení pásu . . . . .	74
4.9	Časový průběh složek reakce $R_L$ . . . . .	74
4.10	Časový průběh složek reakce $R_P$ . . . . .	75
4.11	Zatížení čepu . . . . .	75
4.12	Zatížení čepu II . . . . .	76
4.13	Průběh deformace geometrie pásu . . . . .	78
4.14	Časový průběh výchylky středu pásu . . . . .	79
4.15	Časový průběh rychlosti středu pásu . . . . .	79
4.16	Časový průběh zrychlení středu pásu . . . . .	80
4.17	3D výpočetní model s kombinovaným modelem pásu . . . . .	81
4.18	Sklon pásu . . . . .	82
4.19	Definice okrajové podmínky . . . . .	82
4.20	Časový průběh zrychlení sedadla . . . . .	83

4.21	Diskretizovaný model pásu . . . . .	83
4.22	Reakce v kotevních bodech pásu . . . . .	86
4.23	Směr působení reakcí . . . . .	86
4.24	Směr působení reakcí II . . . . .	87
4.25	Směr působení reakcí III . . . . .	87
4.26	Průběh simulace . . . . .	88
5.1	Reakce v kotevních bodech pásu - 2D analýza . . . . .	90
5.2	Průběh deformace geometrie pásu . . . . .	90
5.3	Reakce v kotevních bodech pásu - 3D analýza . . . . .	91
5.4	Průběh simulace . . . . .	91
5.5	Srovnání průběhů reakcí v kotevních bodech bezpečnostního pásu . . . . .	92



# Seznam tabulek

2.1	Souřadnice materiálové charakteristiky nylonového pásu . . . . .	26
2.2	Souřadnice materiálové charakteristiky polyesterového pásu . . . . .	27
2.3	Materiálové konstanty . . . . .	27
4.1	Rozměry modelu sedadla . . . . .	68
4.2	Rozložení hmotnosti figuríny . . . . .	71
5.1	Dosažené maximální hodnoty reakcí v kotevních bodech bezpečnostního pásu	93



## Příloha G

# Seznam použitých zkratk

**SAE** Society of Automotive Engineers

**LSTC** Livermore Software Technology Corporation

**ATD** Anthropomorphic Test Device

**MKP** Metoda konečných prvků





# Literatura

- [1] ABRATE, S. a C. T. SUN. Dynamic analysis of geometrically nonlinear truss structures. *Computers & Structures*. West Lafayette, IN, USA, 1983, 17(4), 491-497.
- [2] BATHE, Klaus-Jürgen. *Finite element procedures*. Upper Saddle River: Prentice Hall, c1996. ISBN 0-13-301458-4.
- [3] CRISFIELD, M. A. *Non-linear finite element analysis of solids and structures*. New York: Wiley, 1997. ISBN 047195649X.
- [4] DHOLE, Nilesh E. *Development and validation of a finite element model of a transport aircraft seat under part 25.562 dynamic test conditions*. Wichita, KS, USA, 2010. Diplomová práce. Wichita State University.
- [5] DHOLE, Nilesh, Vikas YADAV a Gerardo OLIVARES. Certification by analysis of a typical aircraft seat. *12th International LS-DYNA users conference*. USA, 2012, 12.
- [6] DOYLE, James F. *Nonlinear analysis of thin-walled structures: statics, dynamics, and stability*. New York: Springer, c2001. ISBN 0-387-95216-0.
- [7] GAVIN, H. P. CE 541. Structural dynamics. *Structural elements: Stiffness, mass and damping matrices*. USA, 2014, Durham, NC, USA, 2014. Syllabus. Duke university
- [8] GOWDY, Van, Mark GEORGE a G. A. MCLEAN. Comparison of Buckle Release Timing for Push-Button and Lift-Latch Belt Buckles. Oklahoma City, OK, USA: FAA Civil Aeromedical Institute, 1999.
- [9] JANKOWSKI, Krzysztof *Dynamics of double pendulum with parametric vertical excitation*. Lodz, Polsko, 2011. Diplomová práce. Technical University of Lodz.
- [10] NICKELL, R.E. Nonlinear dynamics by mode superposition. *Computer methods in applied mechanics and engineering*. Albuquerque, NM, USA, 1975, 1976(7), 107-129.
- [11] ONDROUCH, Jan a Jiří PODEŠVA. *Aplikovaný mechanik jako součást týmů konstruktérů a vývojářů: teorie a příklady k předmětu Technické kmitání* [online]. Vyd. 1. Ostrava: VŠB - Technická univerzita Ostrava, 2012 [cit. 2016-03-30]. ISBN 978-80-248-2762-9.
- [12] WRIGGERS, Peter. *Nichtlineare Finite-Element-Methoden*. 2. Dr. Berlin: Springer, 2001. ISBN 9783540677475.

- [13] WU, Shen R. a Lei GU. *Introduction to the explicit finite element method for nonlinear transient dynamics*. Hoboken, N.J.: Wiley, 2012. ISBN 978-0-470-57237-5
- [14] *LS-DYNA: Theory manual*. Livermore software technology corporation, Livermore, California, USA, 2015.
- [15] *LS-DYNA: Keyword user's manual - volume I*. Livermore software technology corporation, Livermore, California, USA, 2015.
- [16] *LS-DYNA: Keyword user's manual - volume II, material models*. Livermore software technology corporation, Livermore, California, USA, 2015.
- [17] *LSTC Hybrid III 50th Fast Dummy: Positioning & Post-Processing*. Livermore software technology corporation, Livermore, Michigan, USA, 2011.
- [18] Mechanical Properties of Plastic Materials. *Professional plastics* [online]. Fullerton, USA, 2016 [cit. 2016-05-11]. Dostupné z: <http://www.professionalplastics.com/professionalplastics/MechanicalPropertiesofPlastics.pdf>
- [19] *SAE AS 8049C: Performance Standard for Seats in Civil Rotorcraft, Transport Aircraft, and General Aviation Aircraft*. SAE International, 2015.
- [20] *Humanetics: Innovative Solutions*. [online]. Plymouth, MI, USA, 2015 [cit. 2016-03-03]. Dostupné z: <http://www.humaneticsatd.com/>.
- [21] *Dummy*. In: The New York Times Magazine [online]. © 2016 The New York Times Company. [vid. 12.3.2016]. Dostupné z: <http://www.nytimes.com/2012/05/20/magazine/>
- [22] *High Strength Aluminium Airplane Buckle*. In: Airplane Seatbelt Extender [online]. © 2015 Airplane Seatbelt Extender. [vid. 3.4.2016]. Dostupné z: <http://www.airplaneseatbelttextender.net/>
- [23] *Aircraft Seatbelt Buckle*. In: National Aircraft Interiors [online]. © 2013 National Aircraft Interiors. [vid. 3.4.2016]. Dostupné z: <http://nat.co.nz/seatbelts.php/>
- [24] *Business-jet-restraint*. In: AmSafe [online]. © 2015 AmSafe. [vid. 3.4.2016]. Dostupné z: <http://amsafe.ajawebtesting.co.uk/restraints/business-jet-restraints/>
- [25] *Rotary Buckle*. In: TRS Motorsport [online]. © 2016 TRS Motorsport Limited. [vid. 3.4.2016]. Dostupné z: [http://www.trs-motorsport.com/harnesses-fia-approved-harnesses-hans-only-harnesses/trs-harnesses/magnum-superlite-6-point-hans-only\\_ct499bd190pd1652.htm/](http://www.trs-motorsport.com/harnesses-fia-approved-harnesses-hans-only-harnesses/trs-harnesses/magnum-superlite-6-point-hans-only_ct499bd190pd1652.htm/)
- [26] *Schroth 4-point restraint system with rotary buckle*. In: Takata Protection Systems [online]. © 2014 Takata Protection Systems. [vid. 3.4.2016]. Dostupné z: <http://takataprotectionsystems.com/general/>

## Poznámky